IFC Release 1.5 Issues/Resolutions Database

Issue Number	***GI - 001***			***Issue Date***	7/8/97
Author	See	***Owner***	See	***Status***	Deferred to R2.0
Schema	All Schemata	***Version***	R1.5 - Pre-Beta		

Issue Description Recent changes are so broad that it is clear we are not even close to stabilization.

Proposed Solution We need to begin using a well disciplined methodology for affecting any and ALL changes to the models. This means first finding a baseline definition for each schema and then agreeing a process for any changes to be made after that. Thomas has suggested a "Change Proposal" system. If we do this, then we will need to expand our STF DB to include tracking of such proposals and references between issues and proposed/completed changes.
Examples:
- Addition of IfcSequence, IfcPlacement, IfcConstructionAid, IfcControl all on the first page of the Kernel since the last STF meetings.
- Subtyping all of the pre-defined properties from the runtime defined IfcPropertyDef (please see notes below in IfcPropertyDefResource).

Resolution Not resolved in first pass (21-Aug-97)

Resolution (25-April-98) will use combination of IRD + FoxPro based tools for this in R2.0.

Action # 1 ***Assignee*** See ***Status*** Eliminated ***Resolved in Version*** R2.0 - Beta

RS and TL will work out a process and make a simple proposal for remainder of R1.5. A more complete proposal to be done for the R2.0 timeframe -- see action 3 from this issue. Simple proposal is to use this tool to track actions. NO CHANGES TO SCHEMATA WITHOUT RECORDING ISSUE AND RESOLUTION IN THIS DB. Confirmed (RS)

Action # 2 ***Assignee*** Liebich ***Status*** Eliminated ***Resolved in Version*** R2.0 - Beta

RS and TL will work out a process and make a simple proposal for remainder of R1.5. A more complete proposal to be done for the R2.0 timeframe -- see action 3 from this issue. Simple proposal is to use this tool to track actions. NO CHANGES TO SCHEMATA WITHOUT RECORDING ISSUE AND RESOLUTION IN THIS DB. Confirmed (RS)

Action # 3 ***Assignee*** See ***Status*** Complete ***Resolved in Version*** R2.0 - Beta

RS will add to list of projects for R2.0 -- A more complete proposal to be done for the R2.0 timeframe -- see action 3 from this issue.

Action # 4 ***Assignee*** See ***Status*** Incomplete ***Resolved in Version*** R2.0 - Beta

Work with Jiri to document process for review by STF

Action # 5 ***Assignee*** Hietanen ***Status*** Incomplete ***Resolved in Version*** R2.0 - Beta

Work with RS to document process for review by STF

Issue Number	***GI - 002***			***Issue Date***	7/8/97
Author	See	***Owner***	Liebich	***Status***	Resolved
Schema	All Schemata	***Version***	R1.5 - Pre-Beta		

Issue Description We have a LOT of schemas for such a "simple" model (relative to the scope we will face in R2,3,4). We now have 16 schemas and 2 more on "gray pages"

Proposed Solution Initially, I would suggest the following simplifications:
- ShapeRep is just another property and could be combined into the Properties Res. This would also address the issue about the TypeDef defined, but not available to ShapeRep
- Construction Aids might be combined with Modeling Aids into a general Utilities/Aids Res. When we introduce it (where did this one come from anyway?). It is driven by requirements in R1.0? -- I understand from Thomas that this has not been absorbed into the Kernel -- right?

Resolution Agreed:

Action # 1 ***Assignee*** Liebich ***Status*** Complete ***Resolved in Version*** R1.5 - Pre-Fin

TL will attempt to subtype IfcShapeRep from IfcPropertyDef (and check consequences). This means that the we will eliminate the ShapeDef schema. Confirmed (RS). Note: IfcPropertyDef name changed to IfcProperty.

Action #	2	***Assignee***	Wix	***Status***	Complete	***Resolved in Version***	R1.5 - Pre-Fin

JW - ConstructionAids was renamed to IfcResource (used in IfcResourceUse by IfcWorkTask). Confirmed (RS).

Issue Number ***GI - 003*** ***Issue Date*** 7/8/97

Author	See	***Owner***	Liebich	***Status***	Resolved
Schema	All Schemata	***Version***	R1.5 - Pre-Beta		

Issue Description Handling of the Root differently in Kernel/ Relationships and Properties - what has been done is not consistent with the 'Pseudo Model' (not using the term 'Meta-Model' here as we have been using that to refer to the SDAI based model definition repository) developed together on 30-May. Either we all need to agree a new meta-model or we need to discuss these inconsistencies (please see notes below in Kernel and PropertyDefRes).

Proposed Solution Implement root info consistently or change the Pseudo Model -- Note: it needs to be updated anyway.

Resolution This was resolved by:

1) the rename of IfcKernelRoot to IfcRoot

2) creation of IfcSeed (includes OwnerID and AuditTrail) which is used in 4 places

3) use of ProjectUniqueID in MANY places

Action #	1	***Assignee***	Liebich	***Status***	Complete	***Resolved in Version***	R1.5 - Pre-Fin

TL will make changes. Confirmed (RS. Note: IfcSeed was eliminated in favor of making AuditTrail an attribute on IfcOwnerHistory (renamed from IfcOwnerID) - which means that IfcOwnerHistory can be used instead of IfcSeed.

Issue Number ***GI - 004*** ***Issue Date*** 7/8/97

Author	See	***Owner***	See	***Status***	Rejected
Schema	All Schemata	***Version***	R1.5 - Pre-Beta		

Issue Description We REALLY NEED TO get some internal documentation into the EXG models. Some of the abstracted relationships and generalizations are very difficult to figure out without documentation that is local to the tool. I know that Jeff started to do this for the IfcPropertyResource.

Proposed Solution We should assign ourselves the task of doing this for all of the schema going forward.

Complication: The only obvious issue is that we need a way to capture this such that it can be regenerated by the tool we use to produce the EXG diagrams after we move onto the Meta-Model toolset.

Resolution Deferred until R2.0 -- new processes for model development.

Rejected for R2.0 -- cannot find a way to do this in an automated way.

Action #	1	***Assignee***	See	***Status***	Complete	***Resolved in Version***	R2.0 - Alpha-1

RS will add to list of projects for R2.0

Issue Number ***GI - 005*** ***Issue Date*** 7/12/97

Author	See	***Owner***	See	***Status***	Deferred to R2.0
Schema	All Schemata	***Version***	R1.5 - Pre-Beta		

Issue Description There are a number of cross-schema issues in this review that will have a significant impact on the toolboxes being built by Concad and CSTB.

Proposed Solution Consider: we may want to advise that they wait until all of the cross schema issues are resolved.

Resolution This is resolved in the EXPRESS code posted in early August -- may still exist in the EXPRESS-G because these two are now disjoint. Issue of coordination of EXPRESS and EXPRESS-G deferred to R2.0.

See I-345 regarding resolution of keeping the EXPRESS-G in sync with EXPRESS - by automating the generation of the EXPRESS-G diagrams.

Need to confirm that R2.0 "Official" EXPRESS code will be issued as "Short Form"

Action # 1 **Assignee** See **Status** Eliminated **Resolved in Version** R2.0 - Alpha

RS: log an issue with regard to toolset - EXPRESS and EXPRESS-G disjoint -- need to generate the EXPRESS-G from the repository based tools or using the STEP Tools generation from EXPRESS.

Action # 2 **Assignee** Liebich **Status** Incomplete **Resolved in Version** R2.0 - Alpha

Confirm publically that R2.0 EXPRESS code will be in Short Form.

Note: has Concad fixed their limitation which made this a problem for them?

Issue Number ***GI - 006*** **Issue Date** 7/12/97

Author See **Owner** See **Status** Rejected

Schema All Schemata **Version** R1.5 - Pre-Beta

Issue Description TypeDefinition -- the enhanced schema is more flexible in that it provides for nesting of TypeDefs (I think) and overriding of individual attributes (something I am not sure our users will want). I have also proposed a slight enhancement that will allow use of multiple typedefs, from differing industry perspecitves (JIM F. -- we talked about this one sometime back). I am somewhat concerned that we may have gone too far with this flexibility and that things will become ambiguous.

Proposed Solution Consider: To know, we need some prototyping and hands-on experience. However, we should be thinking of a logical fallback, just in case.

Resolution Not resolved in first pass (21-Aug-97)

Rejected because this is not specific enough.

Action # 1 **Assignee** See **Status** Eliminated **Resolved in Version** R1.5 - Final

RS: re-submit more specific recommendation --

resolved by other resolutions ..

Issue Number ***GI - 007*** **Issue Date** 7/12/97

Author See **Owner** Liebich **Status** Resolved

Schema IfcPropertyResource **Version** R1.5 - Pre-Beta

Issue Description Subtyping all Properties from IfcPropertyDef -- This is both disturbing and exciting to me. On the one hand, pre-defined simple attributes carry the overhead (and confustion) of the optional PropertyDescriptor (proposed below) and OccurrenceReference -- this is disturbing. On the other hand, this opens up the possibility of attaching ALL attributes at runtime (even predefined ones) and maybe (someday) objects that can change class at runtime. This would be ULTIMATE flexibility -- this is the exciting part. In general, this is contributing to my concern that we are making things WAY to flexible and that performance in implementation will be unacceptable.

Proposed Solution Consider: We need to simplify, simplify, simplify ? even if it means we lose some flexibility.

Resolution Resolved --

1) moved the descriptor from IfcPropertyDef to SimpleProperty and PropertySet (which solves the overhead problem)

2) overriding of attributes has been eliminated

3) subtyping pre-defined properties from IfcPropertyDef will remain -- since #1 above addressed the main concern

Action # 1 **Assignee** See **Status** Complete **Resolved in Version** R1.5 - Pre-Fin

TL will make changes. Confirmed (RS).

Issue Number ***GI - 008*** **Issue Date** 8/8/97

Author See **Owner** All STF **Status** Deferred to R3.0

Schema	All Schemata	**Version**	R1.5 - Pre-Beta

Issue Description Objectified Relationships: I would make the case that Relationships can/should be thought of as typed. If you look at what has been happening to the models in the past 6 weeks, there are a growing number of objectified relationships that are driven simply by associated data. TypeDefinitions were developed to remedy this 'class explosion' and they can be applied equally to objectified relationships as they have been to products. Examples of classes that could be elimintated --> IfcRelUsesProducts, IfcRelUsesConstructionAids, IfcRelConnectsElements, IfcRelGroupsWorks, IfcRelVoidsElements, IfcRelFillsElements, IfcRelAssemblesElements, IfcRelSeparatesSpaces, IfcRelCoversBldgElements, IfcRelGroupsCostSchedules, IfcRelGroupsSpaceProgrammes <-- 11 classes which currently do nothing more than redeclare the relationships (RelatingObject / RelatedObjects).

Proposed Solution 1) add a mandatory attribute "L[0:N] TypeDefinition" [IfcTypeDefResource.IfcPropertyTypeDef] {{ note: this matches the modified attribute recommended for IfcObject}} . 2) add a mandatory attribute "OccuranceProperties" L[0:N] -- as on IfcObject.

Resolution Related to 9 and 10. Not resolved for R1.5 --> deferred to R2.0

Agreed that this is something to consider, but probably too complex for implementers (and STF!) in the R2.0 timeframe. Will look at the possibilities again in the R3.0 timeframe.

Action # 1 **Assignee** Wix **Status** Complete **Resolved in Version** R2.0 - Alpha

JW - Wall Paper view of models (will ask Japanese chapter, who did one for BCCM)

Action # 2 **Assignee** Forester **Status** Complete **Resolved in Version** R2.0 - Alpha

JF - Entity Hierarchy chart

Action # 3 **Assignee** See **Status** Complete **Resolved in Version** R2.0 - Alpha

RS - Long form presentation format for Entities (which shows attr/rela) for each level of Supertypes
(RS will prototype this for a few classes)

Issue Number	***GI - 009***			**Issue Date**	8/8/97
Author	See	**Owner**	All STF	**Status**	Rejected
Schema	All Schemata	**Version**	R1.5 - Pre-Beta		

Issue Description We need to enable redeclaration of objectified relationships w/o creating new classes -- we currently have a REAL BIG problem building in that we have some VERY generalized concepts for which 1) relationships should be redeclared in specializations in order to insure consistent semanic interpretation, however 2) doing so in cases where no additional data/relationships/behavior is defined results in a subtyped class explosion which bloats the model just for the sake of interpretation.

Proposed Solution we need to find a way to provide such redeclaration and/or specialized interpretation of generalized concepts (e.g. RelatingObject/RelatedObjects for Obj.Relationships) without having to create subtyped classes.

Resolution Related to 8 and 10.

The specialized relationships are justified because they have specific target objects and related data. It is also felt that these will include specialized behavior in applications.

Action # 1 **Assignee** See **Status** Complete **Resolved in Version** R2.0 - Alpha

RS and TL will look into a standard way to handle this. 7-Sep-97: RS to include a proposal for this in his proposal for documenting superclasses and inherited interfaces.

Action # 2 **Assignee** Liebich **Status** Eliminated **Resolved in Version** R2.0 - Alpha

RS and TL will look into a standard way to handle this. 7-Sep-97: RS to include a proposal for this in his proposal for documenting superclasses and inherited interfaces.
This has been resolved by the new modeling rule that we will not subtype from concrete Objectified Relationships.

Issue Number	***GI - 010***			**Issue Date**	8/8/97
Author	See	**Owner**	All STF	**Status**	Deferred to R2.0
Schema	All Schemata	**Version**	R1.5 - Pre-Beta		

Issue Description Redeclaration of the relationships on Objectified Rels (for specializations) would be significantly enhanced if we renamed the realtionships to be semanically accurate rather than redeclaring the 'RelatingObject' and 'RelatedObjects' from the abstract level.

Proposed Solution If we have to redeclare anyway, then use semantically accurate relationship names. This may not be allowed in EXPRESS. If not, then we need to find a way to alias the attribute name because it is exceptionally confusing the way it is now (where all redelarations are the same; yet the data types change).

Resolution Related to GI-8 and GI-9
Redeclaration with a changed name cannot be done in EXPRESS.
However, redeclaration can be avoided if we remove the relationship (Relating and Related Objects) in the abstract supertypes - IfcRelationship1to1 and IfcRelationship1toN. See resolution to I-310

Action # 1 ***Assignee*** See ***Status*** Eliminated ***Resolved in Version*** R2.0 - Alpha

RS and TL will look into a standard way to handle this. 7-Sep-97: RS to include a proposal for this in his proposal for documenting superclasses and inherited interfaces.

Action # 2 ***Assignee*** Liebich ***Status*** Eliminated ***Resolved in Version*** R2.0 - Alpha

RS and TL will look into a standard way to handle this. 7-Sep-97: RS to include a proposal for this in his proposal for documenting superclasses and inherited interfaces.

Issue Number GI - 011 ***Issue Date*** 8/8/97

Author See ***Owner*** All STF ***Status*** Deferred to R2.0

Schema All Schemata ***Version*** R1.5 - Pre-Beta

Issue Description We need a way of declaring the semantics of inherited attributes (as well as relationships -- see above). For example: IfcElement.calcTotalArea = "AreaPerSide" for IfcWall, IfcFloor, IfcRoofslab. This can be a REAL problem as our hierarchy gets to be deep because attributes defined in the abstraction layers can be interpreted differently the further removed they are from a leaf class.

Proposed Solution Add an "Attributes and Relationships Re-definition" section to our documentation template -- which only includes redefinition for the ones deemed ambiguous. These can also be filled in over time as we 'discover' which things were ambiguous.

Resolution Recommendation is to create a tool that allows us to declare a more accurate name at the local level -- expanded view of inherited attributes and relationships as described in GI-8.

Cannot do this in time for R1.5. Deferred to R2.0.

Resolution for R2.0 is to capture redefinitions in the model repository and include these specialized semantic defintions in the Class section for the subtype.

Action # 1 ***Assignee*** See ***Status*** Complete ***Resolved in Version*** R2.0 - Alpha

Add to the list of projects for R2.0

Action # 2 ***Assignee*** Hietanen ***Status*** Incomplete ***Resolved in Version*** R2.0 - Alpha

Prototype HTML documentation which presents the specialized semantic definition for an inherited attribute in the Class section for a subtype.
Work w/ TL and implementers on formatting for both the online and HTML documentation.

Action # 3 ***Assignee*** Liebich ***Status*** Incomplete ***Resolved in Version*** R2.0 - Alpha

Work with JH and implementers to define best format to insure use of specialized semantic definitions in both the onlline and printed forms of reference docs.

Issue Number GI - 012 ***Issue Date*** 8/8/97

Author See ***Owner*** All STF ***Status*** Resolved

Schema All Schemata ***Version*** R1.5 - Pre-Beta

Issue Description "Geometry Use" sections of the reference documentation are not yet specific enough. I have received multiple calls complaining that the current scheme in R1.5 allows any object to have ANY shape -- and that this will bring about pandamonium.

Proposed Solution These reference documentation sections should be expanded to define three things which are not currently clear: 1) Standard ShapeRepresentation -- what is the standard use of geometry, 2)

Multiple possible ShapeReps -- where multiple 'standard' possibilities exist, 3) DisAllowed ShapeReps -- where certain use cases are not to be allowed (e.g. use of them will fail certification). This will take a lot of time and cannot be done in a single issue of the IFCs. However, we should state our intention to do so and explain that this clarification will be developed over the next 2 or 3 releases.

Resolution Not resolved in first pass (21-Aug-97).

Fundamentally agreed. However, we will not be able to complete these all in time for R1.5. We will get started and do _some_ in R1.5. Will work to complete for all Class/types which use Implicit Geometry by R2.0.
Final Resolution: Will make use of diagrams from R1.0 and from Implementers agreements. Those not complete will be added to the list of projects for R2.0. Will do #1 for all, #2 for some critical ones for Addendum. All will be done for R2.0.

Action # 1 ***Assignee*** Liebich ***Status*** Incomplete ***Resolved in Version*** R1.5 - Addend

Will do #1 for all, #2 for some critical ones for Addendum

Action # 2 ***Assignee*** See ***Status*** Incomplete ***Resolved in Version*** R2.0 - Alpha

Create list of those not done for R1.5 and put in list of projects for R2.0

Issue Number ***GI - 013*** ***Issue Date*** 8/21/97

Author Wix ***Owner*** All STF ***Status*** Resolved

Schema All Schemata ***Version*** R1.5 - Pre-Beta

Issue Description Aggregate relationships are defined differently thorugh the models

Proposed Solution All 1toN relationships (simple, not objectified) should be declared as mandatory with a minimum low bound of zero

Resolution Just say yes -- do it!

Action # 1 ***Assignee*** Forester ***Status*** Complete ***Resolved in Version*** R1.5 - Pre-Fin

All to revise their schemata to comply with this agreed model design rule.

Action # 2 ***Assignee*** Liebich ***Status*** Complete ***Resolved in Version*** R1.5 - Pre-Fin

All to revise their schemata to comply with this agreed model design rule.

Action # 3 ***Assignee*** See ***Status*** Complete ***Resolved in Version*** R1.5 - Pre-Fin

All to revise their schemata to comply with this agreed model design rule.

Action # 4 ***Assignee*** Wix ***Status*** Complete ***Resolved in Version*** R1.5 - Pre-Fin

All to revise their schemata to comply with this agreed model design rule.

Issue Number ***GI - 014*** ***Issue Date*** 8/21/97

Author Liebich ***Owner*** All STF ***Status*** Resolved

Schema All Schemata ***Version*** R1.5 - Pre-Beta

Issue Description Materials are referenced at very different levels of the model within different branches

Proposed Solution Look to insure consistency in the level at which Materials are referenced

Resolution Resolved by resolutions to other issues.

Issue Number ***GI - 015*** ***Issue Date*** 9/18/97

Author See ***Owner*** See ***Status*** Resolved

Schema All Schemata ***Version*** R1.5 - Pre-Final

Issue Description Model Design Conventions: Naming conventions for Defined data types: All Enumerations should end with "Enum", all Select Types should end with "Select".

Proposed Solution Change the names of the following for the final:
- IfcProfilePreference -- to -- IfcProfilePreferenceEnum

- IfcReferencePreference -- to -- IfcProfilePreferenceEnum
- IfcTransitionCode -- to -- IfcTransitionCodeEnum
- IfcTrimmingPreference -- to -- IfcTrimmingPreferenceEnum
- IfcActor -- to -- IfcActorSelect
- IfcRole -- to -- IfcRoleEnum
- IfcCostOperator -- to -- IfcCostOperatorEnum
- IfcModifierBais -- to -- IfcModifierBaisEnum

Resolution Will do this for all entities that WE define, but will NOT do it for Geometry -- in order to maintain compatibility with STEP part 42.

Action # 1 ***Assignee*** Wix ***Status*** Complete ***Resolved in Version*** R1.5 - Final

Modify names of Enums and Select types accordingly in IfcPropertyResource.

Issue Number I - 001 ***Issue Date*** 7/8/97

Author See ***Owner*** Liebich ***Status*** Resolved

Schema IfcGenericResource ***Version*** R1.5 - Pre-Beta

Issue Description Class: IfcOwnerIdentification.OwningActor - I think it would be useful to create a registry of project team members in the same way we have created a registry of applications which touch the project? In fact, it could be useful in incorporating a model for standard roles for project processes (e.g. workflow control). This would allow application developers to incorporate workflow messaging (e.g. Architect reaches "Arch. Concept Design" milestone and submits to shared model with messages to "Structural Engr" and "HVAC Engr" project roles that they are next in line to create their correstponding "Concept Design"s. This messaging could then be routed to the appropriate team member -- based on who has been assigned these roles in the Project Team Registry. NOTE: I am not suggesting that we include workflow features in R1.5 or even in R2.0, but that a project team registry would be essential to such things in the future, so let's structure for it now and not have to re-structure later.

Proposed Solution OwningActor should be of type INTEGER -- an index into the IfcProjectTeamRegistry - type List[0:N] Ref [IfcActor]. Include a "role" for each actor in the team registry and think about how this could be used for workflow management within the design team. Note: this is different than the document oriented workflow done by products like WorkCenter -- this is workflow in the design process - independent of particular documents.

Resolution TL - The idea of a registry is convincing for both actor and application registry.
Rich: do you volunteer to help defining the correct nice model equivalent?

21-Aug-97 --> compromise seems to be a simple registry of Actors (IfcActorRegistry) and leave the roles and workflow issues to later (maybe R2.0).

Partially resolved -- partially deferred - see I-191

Action # 1 ***Assignee*** Liebich ***Status*** Complete ***Resolved in Version*** R1.5 - Pre-Fin

TL and RS to develop - TL to include this in the UtilitiesResource (renamed from GenericResource). Confirmed (RS).

Action # 2 ***Assignee*** See ***Status*** Complete ***Resolved in Version*** R1.5 - Pre-Fin

TL and RS to develop - TL to include this in the UtilitiesResource (renamed from GenericResource). Confirmed (RS).

Issue Number I - 002 ***Issue Date*** 7/8/97

Author See ***Owner*** Liebich ***Status*** Resolved

Schema IfcGenericResource ***Version*** R1.5 - Pre-Beta

Issue Description Class: IfcOwnerIdentification.UsedApplication is misleading name choice as there will be many users of an object, but only one owner (at any one time).

Proposed Solution "UsedApplication" should be "OwningApplication" .

Resolution Agreed.

Action # 1 ***Assignee*** Liebich ***Status*** Complete ***Resolved in Version*** R1.5 - Pre-Fin

TL to make the change. Confirmed (RS).

Issue Number	***I - 003***			***Issue Date***	7/8/97
Author	See	***Owner***	Liebich	***Status***	Resolved
Schema	IfcGenericResource	***Version***	R1.5 - Pre-Beta		

Issue Description Class: IfcOwnerIdentification.ApplicationRegistry (note spelling) - just a Set of names from an enumeration - this is really ugly. How will we be able to keep a valid list of applications. The original reason for suggesting this was to allow applications which touch the project to register themselves as in the Windows Op. Sys. - NOTE: in that case, Windows does not attempt to maintain an exhaustive list, MS just provides an interface for any app. to register. We should use this model -- it is cleaner and removes the burden of proof from us. FURTHER: if this were a list, then references from OwnerIdent and AudtitTrail could simply use indexes (much more efficient).

Nikolay proposed to add Bentleys products to the list (email 7-Aug-97)

Proposed Solution "OwningApplication" should be of type INTEGER -- an index into the IfcProjectAppRegistry - type List[0:N] Ref [IfcAppIdentification]. IfcAppIdentification should be an class with attributes for: AppFullName: STRING, AppIdentifier: STRING (limited to 8 character), AppDeveloper: IfcActor.

Resolution Agreed.

Action #	1	***Assignee***	Liebich	***Status***	Complete	***Resolved in Version***	R1.5 - Pre-Fin

TL to update EXPRESS per the SS sent by RS. Confirmed (RS).

Issue Number	***I - 004***			***Issue Date***	7/8/97
Author	See	***Owner***	Liebich	***Status***	Resolved
Schema	IfcGenericResource	***Version***	R1.5 - Pre-Beta		

Issue Description Class: IfcAuditTrail.LastModifiedXxx -- Currently this is not a "Trail".

Proposed Solution These 3 attributes should probably be of type - List [1:AuditListLength] -- where "AuditListLength" is another attribute, set by the owning app --> the number of modification records stored in the List.
This idea was pushed by Nikolay in March. I fought it initially as being more complex than we want. His argument was to design it in, even if we force the AuditListLength to 1 for R1.5, R2.0 -- to insure backward compatibility. Complications: The added issue with this is that, to do this "right", we would need to capture a whole lot more information than just "who done it".

Resolution Agreed.

Action #	1	***Assignee***	Liebich	***Status***	Complete	***Resolved in Version***	R1.5 - Pre-Fin

TL to update EXPRESS per the SS sent by RS. Confirmed (RS).

Issue Number	***I - 005***			***Issue Date***	7/8/97
Author	See	***Owner***	Liebich	***Status***	Resolved
Schema	IfcGeometryResource	***Version***	R1.5 - Pre-Beta		

Issue Description Class: IfcAttDrivenRepresentationItem -- Naming problem - not sematically accurate.

Proposed Solution This should really be called IfcAttDrivenGeomRepItem as there are many types of representations besides geometric.

Resolution Will eliminate this supertype and subtype these from IfcGeometricRepresentationItem -- see issue #180.

Action #	1	***Assignee***	Liebich	***Status***	Eliminated	***Resolved in Version***	R1.5 - Pre-Fin

TL to make the change. This change superseded by elimination of this supertype and subtyping the AttDriven types from IfcGeometricRepresentationItem.

Issue Number	***I - 006***			***Issue Date***	7/8/97
Author	See	***Owner***	Liebich	***Status***	Resolved
Schema	IfcGeometryResource	***Version***	R1.5 - Pre-Beta		

Issue Description Class: IfcAttDrivenRepresentationItem -- there is nothing defined for this abstract class!

Proposed Solution	Consider: alternative is to use a SelectType -- what are the consequences?
Resolution	This supertype is now gone as a result of other resolutions.

Issue Number	***I - 007***			***Issue Date***	7/8/97
Author	See	***Owner***	Liebich	***Status***	Deferred to R2.0
Schema	IfcGeometryResource	***Version***	R1.5 - Pre-Beta		

Issue Description	Class: IfcAttDrivenRepresentationItem -- Lost VertexPoint and EdgeCurve as subtypes of GeometricRepresentation. These were useful as topological elements used by connections (for example).
Proposed Solution	Put them back in (please see also comment on Diagram 7 regarding loss of IfcTopologicalRepresentationItems).
Resolution	A proper topological model will be addressed in the R2.0 timeframe.

Action #	1	***Assignee***	See	***Status***	Complete	***Resolved in Version***	R2.0 - Alpha

RS add to list of projects for R2.0

Issue Number	***I - 008***			***Issue Date***	7/8/97
Author	See	***Owner***	Liebich	***Status***	Resolved
Schema	IfcGeometryResource	***Version***	R1.5 - Pre-Beta		

Issue Description	Classes: IfcPlacement and Subtypes (Axis1Placement, Axis2Placement3D, Axis2Placement3D) -- programmer/reader problems in understanding 3 varieties of placement
Proposed Solution	We really need some concept diagrams in order to understand the differences between these 3 types of placement.
Resolution	Descriptions are complete now. Diagrams still need to be added.

Action #	1	***Assignee***	Liebich	***Status***	Complete	***Resolved in Version***	R1.5 - Final

TL will do diagrams

Issue Number	***I - 009***			***Issue Date***	7/8/97
Author	See	***Owner***	Liebich	***Status***	Rejected
Schema	IfcGeometryResource	***Version***	R1.5 - Pre-Beta		

Issue Description	Classes: IfcPlacement and Subtypes (Axis1Placement, Axis2Placement3D, Axis2Placement3D) -- Attribute names like "Z" and "P" are too cryptic.
Proposed Solution	Please use more descriptive names (even if it means they are different than STEP.
Resolution	Rejected. Policy agreed (at this time) is that a STEP entity used exactly 'as is' will keep the attribute names the same.

Action #	1	***Assignee***	See	***Status***	Incomplete	***Resolved in Version***	R2.0 - Alpha

add this to the STF list of policies

Issue Number	***I - 010***			***Issue Date***	7/8/97
Author	See	***Owner***	Liebich	***Status***	Rejected
Schema	IfcGeometryResource	***Version***	R1.5 - Pre-Beta		

Issue Description	Classes: IfcPlacement and Subtypes (Axis1Placement, Axis2Placement3D, Axis2Placement3D) -- Axis1Placement.Axis, Axis2Placement3D.Axis and .RefDirection and Axis2Placement2D.RefDirection all are shown as optional -- how can this be. These objects would be ill-defined without these attributes -- wouldn't they ?
Proposed Solution	make them mandatory.
Resolution	This is consistent with STEP approach -- they assume a default direction if it is not included. Reject change in order to be consistent with STEP entity -- BUT, will issue a SEDS to STEP asking them to change this.

Action # 2 **Assignee** Wix **Status** Eliminated **Resolved in Version** R2.0 - Alpha

TL will write SEDS, JW will push with STEP.

Eliminted: we later discovered that this is solved by the functions in this class.

Action # 1 **Assignee** Liebich **Status** Eliminated **Resolved in Version** R2.0 - Alpha

TL will write SEDS, JW will push with STEP.

Eliminted: we later discovered that this is solved by the functions in this class.

Issue Number I - 011 **Issue Date** 7/8/97

Author See **Owner** Liebich **Status** Resolved

Schema IfcGeometryResource **Version** R1.5 - Pre-Beta

Issue Description Class: IfcCartesianPoint -- Coordinates is shown as a list[1:3] -- seems like this should be [2:3] or even [3:3]. I don't know of a case where we use 1D coordinates, but there are some 2D.

Proposed Solution Coordinates: L[2:3]

Resolution Policy to date has been to 'take it from STEP and apply rules to make acceptable in the IFC context' -- this has been forced to be 2 or 3 through an EXPRESS where rule.

Action # 1 **Assignee** Liebich **Status** Complete **Resolved in Version** R1.5 - Final

TL to add implementers interpretation section to .DOC after the where rules.

Issue Number I - 012 **Issue Date** 7/8/97

Author See **Owner** Liebich **Status** Rejected

Schema IfcGeometryResource **Version** R1.5 - Pre-Beta

Issue Description Classes: IfcCurve and IfcBoundedCurve -- there is nothing defined for this abstract class!

Proposed Solution Consider: alternative is to use a SelectType -- what are the consequences?

Resolution These are needed as they are used as generalizations for data type referenced elsewhere -- leave them in.

Issue Number I - 013 **Issue Date** 7/8/97

Author See **Owner** Liebich **Status** Resolved

Schema IfcGeometryResource **Version** R1.5 - Pre-Beta

Issue Description Class: IfcBoundedCurve -- Error found: 'off page' references for IfcTrimmedCurve and IfcCompositeCurve should be updated to diagram 4 (not 3).

Proposed Solution Fix them

Resolution Fixed in newest

Issue Number I - 014 **Issue Date** 7/8/97

Author See **Owner** Liebich **Status** Rejected

Schema IfcGeometryResource **Version** R1.5 - Pre-Beta

Issue Description Class: IfcLine.Dir -- name "Dir" is misleading -- Vector used defines not only direction, but length as well.

Proposed Solution Dir would better be named "Extent"

Resolution This is consistent with STEP approach attribute naming.

Reject change in order to be consistent with STEP entity -- BUT, will issue a SEDS to STEP asking them to change this.

Action #	1	***Assignee***	See	***Status***	Incomplete	***Resolved in Version***	R2.0 - Alpha
		RS will write SEDS, JW will push with STEP.					
Action #	2	***Assignee***	Wix	***Status***	Incomplete	***Resolved in Version***	R2.0 - Alpha
		RS will write SEDS, JW will push with STEP.					

Issue Number ***I - 015*** ***Issue Date*** 7/8/97

Author See ***Owner*** Liebich ***Status*** Resolved

Schema IfcGeometryResource ***Version*** R1.5 - Pre-Beta

Issue Description Class: IfcVector -- Error found: 'on to page' reference for inheritence should be to Diagram 3 (not 2).

Proposed Solution Fix them

Resolution Fixed in newest

Issue Number ***I - 016*** ***Issue Date*** 7/8/97

Author See ***Owner*** Liebich ***Status*** Resolved

Schema IfcGeometryResource ***Version*** R1.5 - Pre-Beta

Issue Description Class: IfcVector -- Error found: there is a spelling error in the Magnitude reference to IfcMeasureResource.

Proposed Solution Fix them

Resolution Fixed in newest

Issue Number ***I - 017*** ***Issue Date*** 7/8/97

Author See ***Owner*** Liebich ***Status*** Rejected

Schema IfcGeometryResource ***Version*** R1.5 - Pre-Beta

Issue Description Class: Ifc2DCompositeCurve -- there is nothing defined for this class

Proposed Solution Consider: alternative is to use a SelectType -- what are the consequences?

Resolution Rejected. There are 'Where' rules which constrain its use to act in a plane

Issue Number ***I - 018*** ***Issue Date*** 7/8/97

Author See ***Owner*** Liebich ***Status*** Rejected

Schema IfcGeometryResource ***Version*** R1.5 - Pre-Beta

Issue Description Class: Ifc2DCompositeCurve -- the attribute "Outer" : Boolean -- defined for this class in the Alpha-2 review is missing.

Proposed Solution if there is not attribute or relationship for this class, then eliminate it.

Resolution Rejected. This was needed in STEP because it is used with entities that are bounded (where this was set to TRUE), we only use this with unbounded Planes -- therefore we don't need it.

Issue Number ***I - 019*** ***Issue Date*** 7/8/97

Author See ***Owner*** Liebich ***Status*** Resolved

Schema IfcGeometryResource ***Version*** R1.5 - Pre-Beta

Issue Description Class: IfcSolidModel -- there is nothing defined for this abstract class!

Proposed Solution Consider: Subtyping IfcFacetedBrep and IfcSweptAreaSolid -- what are the consequences?

Resolution 21-Aug-97 --> consensus is that we should accept and implement this.

Action #	1	**Assignee**	Liebich	**Status**	Complete	**Resolved in Version**	R1.5 - Pre-Fin

TL to discuss it with Nikolay. This supertype has been eliminated and TL has proposal for combining IfcSweptAreaSolid and IfcAttDrivenExtrusionSolid. See issue on Beta model somewhere after #215.

Issue Number	**I - 020**			**Issue Date**	7/8/97
Author	See	**Owner**	Liebich	**Status**	Resolved
Schema	IfcGeometryResource	**Version**	R1.5 - Pre-Beta		

Issue Description Class: IfcBoundingBox -- The convention for --> where on the box is the origin (or placement) is not clear.

Proposed Solution This must be made crystal clear in documentation.

Resolution This is resolved by the new entity documentation

Issue Number	**I - 021**			**Issue Date**	7/8/97
Author	See	**Owner**	Liebich	**Status**	Resolved
Schema	IfcGeometryResource	**Version**	R1.5 - Pre-Beta		

Issue Description Class: IfcBoundingBox -- The attributes "Z", "Y" and "Z" are not clear and 2 are redundant. Do you mean "X-Dim", "Y-Dim", "Z-Dim" ??

Proposed Solution Eliminate redundancy and make names more descriptive.

Resolution First one was resolved -- error found. Second one agreed -- different than other STEP attribute names policy because this one has a different entity name than the equivalent in STEP.

Action #	1	**Assignee**	Liebich	**Status**	Complete	**Resolved in Version**	R1.5 - Pre-Fin

TL to make changes. Confirmed in Pre-final (RS).

Issue Number	**I - 022**			**Issue Date**	7/8/97
Author	See	**Owner**	Liebich	**Status**	Resolved
Schema	IfcGeometryResource	**Version**	R1.5 - Pre-Beta		

Issue Description Class: IfcClosedShell -- Error found: (2) 'on to page' references should be updated as coming from page 6 (not 5).

Proposed Solution fix them

Resolution Fixed in newest

Issue Number	**I - 023**			**Issue Date**	7/8/97
Author	See	**Owner**	Liebich	**Status**	Rejected
Schema	IfcGeometryResource	**Version**	R1.5 - Pre-Beta		

Issue Description Class: IfcFaceOuterBound -- there is nothing defined for this class!

Proposed Solution Consider: alternative is to use a SelectType -- what are the consequences?

Resolution Rejected -- in favor of STEP compatibility.

Issue Number	**I - 024**			**Issue Date**	7/8/97
Author	See	**Owner**	Liebich	**Status**	Resolved
Schema	IfcGeometryResource	**Version**	R1.5 - Pre-Beta		

Issue Description Class: IfcPolyLoop - Error found: the 'off page' reference to IfcCartesianPoint should be 2,5 (not 1,5)

Proposed Solution fix them

Resolution Fixed in newest

Issue Number	***I - 025***			***Issue Date***	7/8/97
Author	See	***Owner***	Liebich	***Status***	Rejected
Schema	IfcGeometryResource	***Version***	R1.5 - Pre-Beta		

Issue Description The IfcTopologicalRepresentationItems from the Alpha-2 version are missing! These were very useful for connections and alignment of objects. Where have they gone?

Proposed Solution put them back in so that they can be used for alignment and connections based on geometry.

Resolution We decided not to have topological model in R1.5. A proper topological model will be addressed in the R2.0 timeframe. See action on I-7.

Issue Number	***I - 026***			***Issue Date***	7/8/97
Author	See	***Owner***	Liebich	***Status***	Resolved
Schema	IfcGeometryResource	***Version***	R1.5 - Pre-Beta		

Issue Description Class: IfcProfileSegment -- If my assumption about how this works (see question on PathDef below, in IfcAttDrivenPathDef and in ShapeRep schema), the name "IfcProfileSegment" is misleading in that 'Segment' more commonly refers to one segment of a series.

Proposed Solution 'IfcExtrusionSubProfile' would probably be better since it implies that each profile in the list is a subset of the profile 'set' to be extruded along a common path.

Resolution This was a misunderstanding -- these are really segments in a series. However, the IfcProfileSegment is REALLY an ExtrusionSegment.

Action # 1 ***Assignee*** Liebich ***Status*** Complete ***Resolved in Version*** R1.5 - Pre-Fin

TL will change
IfcProfileSegment to IfcExtusionSegment. Confirmed in Pre-final (RS).
IfcStraightSegment to IfcStraightExtrusionSegment. Changed to "UniformExtrusionSegment" (See resolution to I-28). Confirmed in Pre-final (RS).
IfcTaperedSegment to IfcTaperedExtrusionSegment. Confirmed in Pre-final (RS).
IfcMorphingSegment to IfcMorphingExtrusionSegment. Confirmed in Pre-final (RS).

Issue Number	***I - 027***			***Issue Date***	7/8/97
Author	See	***Owner***	Liebich	***Status***	Deferred to R2.0
Schema	IfcGeometryResource	***Version***	R1.5 - Pre-Beta		

Issue Description Class: AttDrivenExtrusionSolid -- Torsion: Boolean -- as per my comments on this 4 months ago, a receiving app cannot do much with the knowledge that an extrusion includes torsion without information defining the rate and direction of torsion -- e.g. 90 degree rotation clockwise about the path for every 5 meters of extrusion.

Proposed Solution Add attributes for rate and direction of torsion

Resolution Torsion will be delayed to R2.0 so that we have more time to resolve the consequences.

Action # 1 ***Assignee*** Liebich ***Status*** Complete ***Resolved in Version*** R1.5 - Pre-Fin

Delete the Torsion attribute for R1.5. Confirmed in Pre-final (RS).
RS will add to projects list for R2.0.

Action # 2 ***Assignee*** See ***Status*** Complete ***Resolved in Version*** R2.0 - Alpha

RS add to list of projects for R2.0

Issue Number	***I - 028***			***Issue Date***	7/8/97
Author	See	***Owner***	Liebich	***Status***	Resolved
Schema	IfcGeometryResource	***Version***	R1.5 - Pre-Beta		

Issue Description Class: IfcStraightSegment -- class name is misleading

Proposed Solution This classname should be 'IfcUniformSubProfile' in that is is not always 'straight' and should be called a SubProfile (rather than segment - see above).

Resolution Uniform" is agreed. 'Sub-profile' was not right -- see last issue.

Action # 1	***Assignee*** Liebich	***Status*** Complete	***Resolved in Version*** R1.5 - Pre-Fin

TL to change "IfcStraightSegment" to "IfcUniformExtrusionSegment". Confirmed in Pre-final (RS).

Issue Number ***I - 029*** ***Issue Date*** 7/8/97

Author	See	***Owner***	Liebich	***Status***	Rejected
Schema	IfcGeometryResource	***Version***	R1.5 - Pre-Beta		

Issue Description Class: IfcTaperedSegment -- Classname is misleading.

Proposed Solution This classname should be 'IfcTaperedSubProfile' (not a segment as explained above).

Resolution Rejected - see I-26

Issue Number ***I - 030*** ***Issue Date*** 7/8/97

Author	See	***Owner***	Liebich	***Status***	Resolved
Schema	IfcGeometryResource	***Version***	R1.5 - Pre-Beta		

Issue Description Class: IfcTaperedSegment -- TaperingFactor: IfcParameterValue - what is this value? Seems too ambiguous.

Proposed Solution define a RateOfTaper: CompoundMeasure (see general notes question above about how to handle 'Unit per Unit' - e.g. Meter (taper) per Meter (of extrusion))

Resolution Should be a ratio. Attribute should be "TaperingRatio" of type IfcRatioMeasure.

Action # 1	***Assignee*** Liebich	***Status*** Complete	***Resolved in Version*** R1.5 - Pre-Fin

TL will make changes. Confirmed in Pre-final (RS).

Issue Number ***I - 031*** ***Issue Date*** 7/8/97

Author	See	***Owner***	Liebich	***Status***	Rejected
Schema	IfcGeometryResource	***Version***	R1.5 - Pre-Beta		

Issue Description Class: IfcMorphingSegment -- Classname is misleading.

Proposed Solution This classname should be 'IfcMorphingSubProfile' (not a segment as explained above).

Resolution Rejected - see I-26

Issue Number ***I - 032*** ***Issue Date*** 7/8/97

Author	See	***Owner***	Liebich	***Status***	Deferred to R2.0
Schema	IfcGeometryResource	***Version***	R1.5 - Pre-Beta		

Issue Description Class: IfcMorphingSegment -- StartProfileDef / EndProfileDef - there appears to be no constraining of these profiles (to be of the same profile type for example --> both rectangular, circular, trapazonidal). This will be a problem if an app defines two different profile types for start/end.

Proposed Solution Constrain these to be of the same profile type and disallow the 'ArbitraryProfile' unless we can constrain the number of verticies to be the same. Additionally, include in the documentation the convention --> that each vertex will map to the like vertex in the next profile (e.g. vertex a-1 extrudes to vertex b-1, etc.).

Resolution Agreed -- and already done --> This is constrained in the 'Where' rules.

Arbitrary profiles and other predefined profiles (ellipse, triangle, etc.) will be considered in R2.0. This may be done with the help of STEP parametric geometry resource.

A method of defining mapping between verticies of dissimilar profiles will also be studied for R2.0.

Action # 1	***Assignee*** See	***Status*** Complete	***Resolved in Version*** R2.0 - Alpha

add to the list of projects for R2.0:
- support of arbitrary and other pre-defined profiles
- method for mapping extrusion from/to verticies of dissimilar profiles

Issue Number I - 033 **Issue Date** 7/8/97

Author See **Owner** Liebich **Status** Rejected

Schema IfcGeometryResource **Version** R1.5 - Pre-Beta

Issue Description Class: IfcAttDrivenProfileDef -- GeometricResolution - the fact that this enumeration allows either 'Curve' or 'Surface' leads me to believe that 'CurveResolution' and 'SurfaceResolution' should be optional (as only one will be used). Right?

Proposed Solution Make them optional (?)

Resolution Rejected. Derived (DER) attributes cannot be optional in EXPRESS (arrrrgh!)

However, some changes were agreed.

Action # 1 **Assignee** Liebich **Status** Complete **Resolved in Version** R1.5 - Pre-Fin

TL to change name of attribute from GeometricResolution to ResultingGeomType, the enum from IfcProfilePreference to IfcSurfaceOrSolid, attribute 'CurveResolution' to 'CurveForSurface', 'SurfaceResolution' to 'SurfaceForSolid'. Note: IfcSurfaceOrSolid is not right for "ResultingGeomType" of a profile -- set to IfcCurveOrSurface (where a profile that is a Curve will be extruded to create a surface and a Surface will be extruded to create a solid. Confirmed in Pre-Final (RS). See also GI-15 for name of Enum

Issue Number I - 034 **Issue Date** 7/8/97

Author See **Owner** Liebich **Status** Resolved

Schema IfcGeometryResource **Version** R1.5 - Pre-Beta

Issue Description Class: IfcCircleProfileDef, IfcRectangleProfileDef, IfcTrapeziumProfileDef -- it is VERY difficult to sleuth what some of the attributes mean without concept diagrams.

Proposed Solution Complete concept diagrams for each of these profiles which show each attribute.

Resolution Cannot use 'Length', 'Width', etc. here because the use of the profile in different cases will be different. Compromise --> "Xdim", "Ydim", etc.

Action # 1 **Assignee** Liebich **Status** Complete **Resolved in Version** R1.5 - Final

TL to make changes.

Issue Number I - 035 **Issue Date** 7/8/97

Author See **Owner** Liebich **Status** Resolved

Schema IfcGeometryResource **Version** R1.5 - Pre-Beta

Issue Description Class: IfcCircleProfileDef, IfcRectangleProfileDef, IfcTrapeziumProfileDef -- Radius, Y, X, BottomX, TopX, Y, MaxX, MaxY - these names are too cryptic!

Proposed Solution Please make the attribute names descriptive - even if it means they are different from STEP -- as they were in the Alpha-2 versions.

Resolution Agreed to do the same as in I-34.

Action # 1 **Assignee** Liebich **Status** Complete **Resolved in Version** R1.5 - Final

TL to make changes

Issue Number I - 036 **Issue Date** 7/8/97

Author See **Owner** Liebich **Status** Deferred to R2.0

Schema IfcGeometryResource **Version** R1.5 - Pre-Beta

Issue Description Class: IfcAttDrivenPathDef -- The mulit-segment paths defined in R1.0 and in the Alpha are missing. These are VERY important and powerful in that it is clear to the receiving application, how to clean up the 'joints'. As you will remember, this was an issue for the implementers at first (in that they had not used a system for unambiguously transferring such connection geometry before), but then became one of the most obvious features of the demos in Frankfurt and Philadelphia.

Proposed Solution Restore muli-segment (BoundedCurve) paths as in R1.0 and Alpha-2

Resolution Convention is that the extrusion is along the 'Z'-axis of the local placement of the Extrusion Segment (see IfcExtrusionSolid). Not resolved in first pass (21-Aug-97). Second pass (23-Aug-97) - We will live with single segment paths for R1.5 -- will look at this again in R2.0.

Action #	1	***Assignee***	See	***Status***	Complete	***Resolved in Version***	R2.0 - Alpha

RS: add to the list of projects for R2.0 --> consider restoring multi-curve extrusion paths (as in R1.0)

Issue Number ***I - 037*** ***Issue Date*** 7/8/97

Author	See	***Owner***	Liebich	***Status***	Rejected
Schema	IfcGeometryResource	***Version***	R1.5 - Pre-Beta		

Issue Description Class: IfcArcPathDef -- Where is the center of the Arc? You have ExtrAngles and Radius, but can't construct the Arc path without a center point.

Proposed Solution Add center of Arc or clarify where it is defined.

Resolution Convention is that the center is the origin of the local coordinate system

However, one change was agreed.

Action #	1	***Assignee***	Liebich	***Status***	Eliminated	***Resolved in Version***	R1.5 - Pre-Fin

TL will add reference to Local Placement on the IfcAttDrivenPathDef. Note confirmed in Pre-Final (RS - email to TL 15-Sep).

Issue Number ***I - 038*** ***Issue Date*** 7/8/97

Author	See	***Owner***	Liebich	***Status***	Rejected
Schema	IfcGeometryResource	***Version***	R1.5 - Pre-Beta		

Issue Description Class: IfcArcPathDef -- ExtrAngles: L[1:N] - why is this a list of angles. You should only need angle to extrude 'from' and angle to extrude 'to'.

Proposed Solution Change to ExtAngleStart and ExtAngleEnd.

Resolution Rejected. This is mis-understood -- this list allows multiple extrusion segment along the curve.

Issue Number ***I - 039*** ***Issue Date*** 7/8/97

Author	See	***Owner***	Liebich	***Status***	Resolved
Schema	IfcGeometryResource	***Version***	R1.5 - Pre-Beta		

Issue Description Class: IfcStraightPathDef -- Where are the starting point and Direction for this path?. How can the receiving system reconstruct the path without these?

Proposed Solution Add starting point and direction (or change ExtrLength to a Vector).

Resolution Rejected. Convention is that the start is the origin of the local placement -- now to be put on the IfcAttDrivenDef

---

Issue Number ***I - 040*** ***Issue Date*** 7/8/97

Author	See	***Owner***	Liebich	***Status***	Deferred to R2.0
Schema	IfcGeometryResource	***Version***	R1.5 - Pre-Beta		

Issue Description Class: IfcStraightPathDef -- ExtrLengths: L[1:N] - why is this a list of lengths. If this is a single segment extrusion (see other notes on this), then only one should be needed. See also the note above on multi-segment paths.

Proposed Solution Change this to a single length for this single segement path definition.

Resolution Not resolved in first pass (21-Aug-97) -- to be resolved with I-36. Second pass (23-Aug-97) - We will live with single segment paths for R1.5 -- will look at this again in R2.0.

Action #	1	***Assignee***	See	***Status***	Complete	***Resolved in Version***	R2.0 - Alpha

RS: add to the list of R2.0 STF projects

Issue Number I - 041 **Issue Date** 7/8/97

Author See **Owner** Liebich **Status** Resolved

Schema IfcShapeRepResource **Version** R1.5 - Pre-Beta

Issue Description General comments - This schema seems too complex. Why does it use two separate levels of containment -- Product and ShapeRep. Introduction of the "Product" terminology here is confusing and foreign to an AEC application developer. Currently we have ProductShape; containing ProductComponentShapes; which contain ShapeReps.

Proposed Solution Why not simply allow nesting of ComponentShapes (components can have components --> to any level of detail) which are contained within a ShapeRepresentation which is referenced as a Property of a semantic model object.

Resolution Nesting agreed. Elimination of Positive/Negative subtypes agreed. Eliminated IfcProductComponentShape (reference IfcShapeRep directly from IfcProductShape).

Add Boolean (PositiveOrNegative) to IfcProductShape (to replace removed subtypes).

Remove TypeDefID (this was added originally to allow PropertySets on ProductShape components = mixing of semantic and geometric models. Remove "Usage" as this in now replaced by "Description" pushed up to ProductShape (from ComponentShape).

Semantic model obj. points to IfcProductShape, which refs List[0:N] IfcProductShape (self reference), which optionally refs IfcShapeRepresentation (optional in the case where the shape is only defined by the component ProductShapes).

Action # 1 **Assignee** Liebich **Status** Complete **Resolved in Version** R1.5 - Pre-Fin

TL to make changes. Note: the nesting proposed has been implemented using a recursive 'CSG-like' tree structure which allow combination of any number of component shapes and the use of boolean operators (limited to subtraction for R1.5). Confirmed in Pre-Final (RS).

Issue Number I - 042 **Issue Date** 7/8/97

Author See **Owner** Liebich **Status** Resolved

Schema IfcShapeRepResource **Version** R1.5 - Pre-Beta

Issue Description Class: IfcProductShape -- ProjectID, OwnerID, AuditTrail - these three are defined in IfcRoot agree in San Rafael on 30-May. They should not be attached independently in multiple places. We agree that the IfcRoot should be defined independently and then contained (aggregated) into three root classes at per our 'Pseudo Model' (see also the discussion in A-2c).

Proposed Solution If we want ID on shape (see next issue), then it should be done through aggregation of a common IfcRoot object.

Resolution 1) IfcKernelRoot will now be IfcRoot and will have a single attribute (IfcProjectUniqueID).
2) IfcSeed will be defined in the GenericResource and will include the IfcOwnerID and the IfcAuditTrail
3) IfcSeed will be contained by IfcObject, IfcRelationship, IfcProject and IfcPropertyTypeDef
4) All objects in a project should reference IfcProjectUniqueID

Action # 1 **Assignee** Liebich **Status** Complete **Resolved in Version** R1.5 - Pre-Fin

TL to make changes. Note: changes since this was captured. 1) IfcSeed is now IfcOwnerHistory. Confirmed with exception - IfcSeed refs in Kernel and IfcPropertyTypeResource should be updated to IfcOwnerHistory (RS email to TL - 15-Sep)

Issue Number I - 043 **Issue Date** 7/8/97

Author See **Owner** Liebich **Status** Rejected

Schema IfcShapeRepResource **Version** R1.5 - Pre-Beta

Issue Description Class: IfcProductShape -- ID on ShapeReps - We did not include the IfcRoot (ID) object in the ShapeRep in our 'Pseudo Model' because we argued that the shape is not independent of the owning object, therefore. We agreed that we have to make some hard choices about which objects need independent ID because we need to reduce the overhead involved in putting this type of 'heavy' identification and tracking on every property in our model. This will be a performance killer.

Proposed Solution Look into the consequences of excluding independent ID on all properties, including ShapeRep. We may find that we have to, but if we don't, then we should try to reduce this overhead.

Resolution Rejected -- see decision #4 on I-42.

Issue Number	**I - 044**			**Issue Date**	7/8/97
Author	See	**Owner**	Liebich	**Status**	Resolved
Schema	IfcShapeRepResource	**Version**	R1.5 - Pre-Beta		

Issue Description Class: IfcProductShape -- MainComponent/SubComponents - I tend to agree with other notes I have seen that this distinction of a main component seems somewhat artificial. I don't see the advantage other than it being viewed as the basis for the additions and subtractions of sub-components (which I don't think we need if we use a LIST of components (#1 in the list becomes the basis).

Proposed Solution Remove the Main/Sub component distinction and allow components to be nested as destibed in the general notes for this schema.

Resolution Resolved -- see solution described in I-41

Issue Number	**I - 045**			**Issue Date**	7/8/97
Author	See	**Owner**	Liebich	**Status**	Resolved
Schema	IfcShapeRepResource	**Version**	R1.5 - Pre-Beta		

Issue Description Class: IfcProductShape -- Usage:STRING - Is this attribute really supposed to be a "Description" of the ProductShape?.

Proposed Solution Pick a more semanically accurate attribute name.

Resolution Resolved -- see solution described in I-41

Issue Number	**I - 046**			**Issue Date**	7/8/97
Author	See	**Owner**	Liebich	**Status**	Rejected
Schema	IfcShapeRepResource	**Version**	R1.5 - Pre-Beta		

Issue Description Class: IfcProductComponentShape -- ProjectId - do we really want to track an ID for every component of every object in our models? This seems like awfully heavy overhead. So far as I can see, these component shapes do not exist independently and are not shared between multiple objects, therefore we should be able to contain them in the owning object instance (which has independent ID).

Proposed Solution Look into the consequences of excluding independent ID for components.

Resolution Rejected -- see decision #4 on I-42.

Issue Number	**I - 047**			**Issue Date**	7/8/97
Author	See	**Owner**	Liebich	**Status**	Resolved
Schema	IfcShapeRepResource	**Version**	R1.5 - Pre-Beta		

Issue Description Class: IfcProductComponentShape -- TypeDefID:STRING - ShapeReps currently don't have TypeDefinitions, so what could this be used for?

Proposed Solution Eliminate this attribute unless we enhance ShapeDefs to allow TypeDefinition -- something I don't think would be very useful.

Resolution Agreed - Resolved in solution described in I-42

Issue Number	**I - 048**			**Issue Date**	7/8/97
Author	See	**Owner**	Liebich	**Status**	Resolved
Schema	IfcShapeRepResource	**Version**	R1.5 - Pre-Beta		

Issue Description Class: IfcPositiveComponentShape / IfcNegativeComponentShape -- So far as I can tell, these two subtypes do nothing.

Proposed Solution Add a LOGICAL attribute on IfcProductComponentShape (or IfcComponentShape as recommended above) which states whether the component shape is positive or negative.

Resolution Agreed - Resolved in solution described in I-42

IFC Release 1.5 Issues/Resolutions Database

Issue Number	***I - 049***			**Issue Date**	7/8/97
Author	See	**Owner**	Liebich	**Status**	Resolved
Schema	IfcShapeRepResource	**Version**	R1.5 - Pre-Beta		

Issue Description Class: IfcRepresentationContext -- IfcRepViewSelect / IfcRepViewType / IfcUserDefinedType - While I believe that it is a good idea to define such "Types" for shape representation now (even though we are only doing physical ShapeRep in R1.5 and R2.0), I do believe that UserDefinedTypes is over the top at this time. Let's just define some standard types for now and SIMPLIFY.

Nikolay seconds this one (7-Aug-97)

Proposed Solution Eliminate IfcRepViewSelect and the reference to IfcMeasureResource.IfcUserDefinedType --> ViewType:IfcRepViewType.

Resolution Agreed:
1) remove IfcRepViewSelect and IfcUserDefinedType
2) directly reference IfcRepViewTypeEnum (note name change) from IfcRepresentationContext and add more types to this enumeration (Plan, Section, Elevation, Isometric, Diagramatic,Undefined)
3) add IfcRepViewDetailEnum which includes (Sketch, Outline, Design, Detail, Undefined)

Action # 1 **Assignee** Liebich **Status** Complete **Resolved in Version** R1.5 - Pre-Fin

TL to make changes. Confirmed in Pre-Final (RS).

Issue Number	***I - 050***			**Issue Date**	7/8/97
Author	See	**Owner**	Liebich	**Status**	Resolved
Schema	IfcShapeRepResource	**Version**	R1.5 - Pre-Beta		

Issue Description Class: IfcRepresentationContext -- Error found: IfcMeasureResource.IfcUserDefinedType does not exist in the .EXG file for the IfcMeasureResource schema.

Proposed Solution fix it

Resolution Already fixed

Issue Number	***I - 051***			**Issue Date**	7/8/97
Author	See	**Owner**	Liebich	**Status**	Deferred to R2.0
Schema	IfcShapeRepResource	**Version**	R1.5 - Pre-Beta		

Issue Description Class: IfcRepresentationContext -- PreferenceType:IfcRepPreferenceType [Accurate, Approximate] - what does this mean -- that the creating app preferred this type of rep or that the associated rep IS Accurate or Approximate?

Proposed Solution Use a more semantically accurate attribute name -- such as "IfcRepresentationAccuracy"

Resolution Eliminate for R1.5 and study for better solution in R2.0.

Action # 1 **Assignee** Liebich **Status** Complete **Resolved in Version** R1.5 - Pre-Fin

1) TL to make the change and communicate with Eberhard M. - why was he arguing for this.   Confirmed in Pre-Final (RS).

Action # 2 **Assignee** See **Status** Complete **Resolved in Version** R2.0 - Alpha

2) RS to add this to the list of STF projects for R2.0

Issue Number	***I - 052***			**Issue Date**	7/8/97
Author	See	**Owner**	Liebich	**Status**	Rejected
Schema	IfcShapeRepResource	**Version**	R1.5 - Pre-Beta		

Issue Description Class: IfcShapeRepresentation -- ProjectId - As with components, I believe we will want to avoid tracking an ID for every ShapeRep forf every object in our models? So far as I can see, these ShapeReps do not exist independently and are not shared between multiple objects, therefore we should be able to contain them in the owning object instance (which has independent ID).

Proposed Solution	Look into the consequences of excluding independent ID for ShapeReps.
Resolution	Rejected - see resolution item #4 on I-42

Issue Number	***I - 053***			***Issue Date***	7/12/97
Author	See	***Owner***	Wix	***Status***	Resolved
Schema	IfcMeasureResource	***Version***	R1.5 - Pre-Beta		

Issue Description	DefinedTypes: IfcCompoundPlaneAngleMeasure and IfcSolidAngleMeasure -- the first of these is new since the Alpha Reviews and is a List of 3 REAL and the second is a single REAL -- Is the first used for Degrees/Minutes/Seconds (=Surveyor's angle measure) and the second is in decimal degrees? If so, I believe the first should be a List of INTEGER as I don't think I have every seen decimal values used in Surveyor's angle measure.
Proposed Solution	use an INTEGER
Resolution	Agreed -- but also have to constrain to list of [3:3] of integer. Also need to enhance documentation to describe where to use each of IfcCompoundPlaneAngleMeasure and IfcSolidAngleMeasure.

Action #	1	***Assignee***	Wix	***Status***	Complete	***Resolved in Version***	R1.5 - Final

JW to make changes described above, plus add to documentation re: where to use each.
Model change confirmed in Pre-Final (RS). Doc change confirmed 26-Nov-97

Issue Number	***I - 054***			***Issue Date***	7/12/97
Author	See	***Owner***	Wix	***Status***	Resolved
Schema	IfcMeasureResource	***Version***	R1.5 - Pre-Beta		

Issue Description	Class: IfcTimeDuration -- this appears to be the replacement for the IfcCompoundTimeDurationMeasure -- this should REMAIN one of the IfcMeasureValue select type choices -- it is a measure of time duration.
Proposed Solution	Include it in the set of IfcMeasureValue possibilities -- cross page ref. from diagram 2 to this entity
Resolution	Agreed, but there is a complication -- all of the MeasureValues are defined data types. Proposed solution: 1) eliminate the IfcTimeDuration class and replaced it with a defined data type of IfcTimeDurationMeasure [REAL], also add time measurement units to the UnitsInContext. 2) Move IfcCalendar, IfcDateAndTime, IfcLocalTime to the IfcPropertyResource schema and subtype each from IfcPropertyDef (so the they are available for use in PropertySets.

Action #	1	***Assignee***	Wix	***Status***	Complete	***Resolved in Version***	R1.5 - Pre-Fin

JW to make changes. Confirmed in Pre-Final (RS).

Issue Number	***I - 055***			***Issue Date***	7/12/97
Author	See	***Owner***	Wix	***Status***	Resolved
Schema	IfcMeasureResource	***Version***	R1.5 - Pre-Beta		

Issue Description	Class: IfcTimeDuration -- EndTime - why is this optional?? It cannot be optional if you are to have a duration because you need two times to do that.
Proposed Solution	make is mandatory
Resolution	Agreed -- resolved in the solution presented in I-54

Issue Number	***I - 056***			***Issue Date***	7/12/97
Author	See	***Owner***	Wix	***Status***	Resolved
Schema	IfcMeasureResource	***Version***	R1.5 - Pre-Beta		

Issue Description	Class: IfcCoordinatedUnniversalTimeOffset -- Sense [EnumeratedType] - Again (see Alpha review notes), I don't see why this is an Enumeration!
Proposed Solution	It can only be ahead or behind, so it should be a boolean called "Ahead". In the case where it is

the same, make it true and set the offset to zero.

Resolution Agreed.

Action # 1 ***Assignee*** Wix ***Status*** Complete ***Resolved in Version*** R1.5 - Pre-Fin

JW to make changes. Not confirmed in Pre-Final (RS- email to JW, 15-Sep).

Issue Number ***I - 057*** ***Issue Date*** 7/12/97

Author See ***Owner*** Wix ***Status*** Rejected

Schema IfcPropertyResource ***Version*** R1.5 - Pre-Beta

Issue Description Superclass: IfcTypeDefResource.IfcPropertyDef -- Subtyping off of IfcPropertyDef is not shown in the IfcTypeDefResource schema.

Proposed Solution Update IfcTypeDefResource schema.

Resolution Rejected -- this is a limitation of the tools we are using -- cannot show inheritence to another schema

---

Issue Number ***I - 058*** ***Issue Date*** 7/12/97

Author See ***Owner*** Wix ***Status*** Rejected

Schema IfcPropertyResource ***Version*** R1.5 - Pre-Beta

Issue Description Class: IfcPersonAnOrganization -- I believe that this class should be eliminated and one attribute added to each of IfcPerson and IfcOrganization.

Proposed Solution The design change proposed will allow everything possible now AND will allow association of multiple persons with an organization (e.g. BuildingAuthority is listed as an Actor and there are 3 plan checkers assigned to this project.

CHANGES PROPOSED:
1) eliminate IfcPersonAndOrganization altogether
2) add an optional attribute "Organization" on IfcPerson
3) add a mandatory attribute "Persons L[0:N] Ref [IfcPerson]".

Resolution Rejected - this does not allow a person to be in multiple organizations.

Issue Number ***I - 059*** ***Issue Date*** 7/12/97

Author See ***Owner*** Wix ***Status*** Rejected

Schema IfcPropertyResource ***Version*** R1.5 - Pre-Beta

Issue Description Class: IfcPerson -- At least one field in every class should be mandatory. In this case it does not make sense to allow a person for which you have no name.

Proposed Solution Make FamilyName and GivenName mandatory.

Resolution Rejected -- There is a 'where' rule which requires one of the two names.

Issue Number ***I - 060*** ***Issue Date*** 7/12/97

Author See ***Owner*** Wix ***Status*** Resolved

Schema IfcPropertyResource ***Version*** R1.5 - Pre-Beta

Issue Description Class: IfcPerson -- MiddleNames, PrefixTitles, SuffixTitles (all L[1:N]). allowing a list for each of these is "over the top" and unnecessary -- since they are STRINGs, a list can (and should) be concatenated.

Proposed Solution Reduce each to a single optional STRING value.

Resolution Agreed.

Action # 1 ***Assignee*** Wix ***Status*** Complete ***Resolved in Version*** R1.5 - Pre-Fin

JW to make changes. Confirmed in Pre-Final (RS).

Issue Number ***I - 061*** ***Issue Date*** 7/12/97

Author	See	***Owner***	Wix	***Status***	Resolved
Schema	IfcPropertyResource	***Version***	R1.5 - Pre-Beta		

Issue Description Class: IfcPerson --Addresses[L[1:N], Roles[L1:N] - Somewhere along the way, we lost our convention to support implementers by eliminating optional Lists and Sets --> in favor of mandatory [0:N].

Proposed Solution Change each of these to mandatory L[0:N].

Resolution Resolved by policy.

Action # 1 ***Assignee*** Wix ***Status*** Complete ***Resolved in Version*** R1.5 - Pre-Fin

JW to make changes. Confirmed in Pre-Final (RS).

Issue Number ***I - 062*** ***Issue Date*** 7/12/97

Author	See	***Owner***	Wix	***Status***	Resolved
Schema	IfcPropertyResource	***Version***	R1.5 - Pre-Beta		

Issue Description Class: IfcOrganization -- Addresses[L[1:N], Roles[L1:N] - Somewhere along the way, we lost our convention to support implementers by eliminating optional Lists and Sets --> in favor of mandatory [0:N].

Proposed Solution Change each of these to mandatory L[0:N].

Resolution Resolved by policy.

Action # 1 ***Assignee*** Wix ***Status*** Complete ***Resolved in Version*** R1.5 - Pre-Fin

JW to make changes. Confirmed in Pre-Final (RS).

Issue Number ***I - 063*** ***Issue Date*** 7/12/97

Author	See	***Owner***	Wix	***Status***	Rejected
Schema	IfcPropertyResource	***Version***	R1.5 - Pre-Beta		

Issue Description Class: IfcAddress -- At least one field in every class should be mandatory. In this case it does not make sense to allow an address for which there is not AT LEAST the Town and Country.

Proposed Solution Make Town and Country mandatory

Resolution Rejected -- There is a 'where' rule which requires one of the attributes..

Issue Number ***I - 064*** ***Issue Date*** 7/12/97

Author	See	***Owner***	Wix	***Status***	Rejected
Schema	IfcPropertyResource	***Version***	R1.5 - Pre-Beta		

Issue Description Superclass: IfcTypeDefResource.IfcPropertyDef --Subtyping off of IfcPropertyDef is not shown in that schema.

Proposed Solution Update IfcTypeDefResource schema

Resolution Rejected -- this is a limitation of the tools we are using -- cannot show inheritence to another schema

---

Issue Number ***I - 065*** ***Issue Date*** 8/8/97

Author	See	***Owner***	Wix	***Status***	Resolved
Schema	IfcPropertyResource	***Version***	R1.5 - Pre-Beta		

Issue Description Class: IfcMaterialLayer - Relationships between the parts in a MaterialLayerSet and its use in an occurrence of Wall, Floor, etc. is VERY confusing.

Proposed Solution Create and include in the documentation the diagram we (STF) drew on the whiteboard on 30-May-97 in San Rafael.

Resolution Agreed.

Action # 1 **Assignee** Forester **Status** Complete **Resolved in Version** R1.5 - Final

JF will create the diagram (from notes during the May STF meeting) and pass to JW.

Action # 2 **Assignee** Wix **Status** Complete **Resolved in Version** R1.5 - Pre-Fin

JW to incorporate the diagram into the documentation.

Issue Number ***I - 066*** **Issue Date** 8/8/97

Author See **Owner** Wix **Status** Resolved

Schema IfcPropertyResource **Version** R1.5 - Pre-Beta

Issue Description Class: IfcMaterialLayer - LayerOffset [IfcLengthMeasure] -- the meaning of this attribute is STILL ambiguous -- even in the .DOC file.

Proposed Solution 1) rename to "OffsetFromMlsBase" (Mls=MaterialLayerSet, "MlsBase" = outside face of Layer 1 (first in list) -- depends on the "Sense" defined in each occurrence (see IfcWall for example), 2) CLEARLY state in the documentation that the offset is from this "MlsBase" to the first face of the layer (layer thickness is always positive and continues in the "Sense" direction to the other layer face) -- NOTE: Positive measure will be taken to mean --> in the direction defined by sense (e.g. a sense of LeftToRight means measure is positive from "Left to Right").

Resolution Agreed.

Action # 1 **Assignee** Wix **Status** Complete **Resolved in Version** R1.5 - Final

JF and JW will implement. Model change confirmed, doc extensions not (15-Sep).

Action # 2 **Assignee** Forester **Status** Complete **Resolved in Version** R1.5 - Pre-Fin

JF and JW will implement. Model change confirmed, doc extensions not (15-Sep).

Issue Number ***I - 067*** **Issue Date** 7/12/97

Author See **Owner** Wix **Status** Deferred to R2.0

Schema IfcPropertyResource **Version** R1.5 - Pre-Beta

Issue Description Class: IfcMaterialLayer -- Material[Ref [IfcMaterial] - I have long been bothered by the fact that our MaterialLayerSets do not handle composite or elemented configurations well. Issue: how do we use this for ElementedWalls? --> e.g. 1) insulated stud wall, 2) concrete wall w/ repeating pilaster.

Proposed Solution Consider: IfcMaterialLayerComposition which provides for the definition of 1) % of phyisical volume filled by alternative materials, 2) spacing (along extrusion path) for repeating elements, 3) length (along extrusion path) for repeating elements --> this would be VERY useful to simulation apps and to CAD apps generating views of such layers.

Resolution This is too complex for R1.5. Delay to projects for R2.0.
(JW-980510) Accepting that this is not a final solution to the question of layering (which will need to be put off to R3 due to current constraints):

Include a new class of IfcMaterialComponent where the material component is manufactured/ constructed from exactly one Material. Make a relationship between IfcMaterialLayer and IfcMaterialComponent such that an IfcMaterialLayer has at least one IfcMaterialComponent (to account for the situation where the layer in fact comprises a single material). Allow for the IfcMaterialComponent to be a placed with an offset from the MLSBase as for the IfcMaterialLayer. It shall also have an XaxisRelOffset and a ZaxisRelOffset as positive length measures so that its location within the layer can be determined. Also allow for the IfcMaterialComponent to have a positive length and height. Width is not specified since, for present purposes, the width of the IfcMaterialComponent should be considered to be the width of the layer that contains it by default.

Action # 1 **Assignee** See **Status** Complete **Resolved in Version** R2.0 - Alpha

Add this to the list of projects for R2.0.

Issue Number ***I - 068*** **Issue Date** 7/12/97

Author See **Owner** Wix **Status** Deferred to R2.0

Schema IfcPropertyResource **Version** R1.5 - Pre-Beta

Issue Description Class: IfcMaterial -- MaterialName [STRING] - using STRINGS for material definition has VERY limited value.

Proposed Solution Consider: references into an industry standard (international?) for construction materials. Short of this (if we cannot find one), it would be MORE USEFUL is we defined an enum of pre-defined materials and an optional STRING to support cases where "Other" is used from the Enum.

Resolution This is too complex for R1.5. Delay to projects for R2.0..
(JW-980510) Considering the Uniclass classification, I see the following main material groups and sub groups

(Material
 (Ston
  (Natur
(Basalt, Bauxite, Chalk, Flint, Granite, Gravel, Gritstone, Limestone, Marble, Quartzite, Sand, Sandstone, Slate)
, Reconstituted
(……..)

 (Cementitious and Concrete and Mineral Bound Material
(Cementitious Materials, Cementitious Binders, Concrete, Other Mineral Bound Materials)

 (Mineral
  (Mineral Based Materials, Soils, Clay Based Materials, Bitumen Based Material

 (Meta
  (Steel, Iron, Aluminium, Copper, Zinc, Lead, Other Meta

 (Timbe
  (General Wood, Laminated Wood, Fibre Building Boar

 (Animal and Vegetable material
  (Animal Material, Vegetable Materia

 (Plastics and Rubber and Chemicals and Synthetic
(Plastics General, Plastics Composite, Natural Rubber, Synthetic Rubber, Synthetic Chemicals)
 (Combined and Undefined Material
  (Composite Material, Othe

)

There are in fact 2 pages of them. We would probably need to add others from other classification systems to cover the range of classifications. The model file C-Uni shows the above as a hierachical subtype model (schema would be identified as Classification-Uniclass or something of that nature; others might be Classification-CISfB, Classification-CAWS, Classification-Masterformat etc.). See remarks against Classification issues for further suggested amendments to the Classification model.

Action # 1 **Assignee** See **Status** Complete **Resolved in Version** R2.0 - Alpha

Add this to the list of projects for R2.0.

Issue Number **I - 069** **Issue Date** 7/12/97

Author See **Owner** Wix **Status** Rejected

Schema IfcPropertyResource **Version** R1.5 - Pre-Beta

Issue Description Superclass: IfcTypeDefResource.IfcPropertyDef -- Subtyping off of IfcPropertyDef is not shown in that schema.

Proposed Solution Update IfcTypeDefResource schema

Resolution Rejected -- this is a limitation of the tools we are using -- cannot show inheritence to another schema

---

Issue Number **I - 070** **Issue Date** 7/12/97

Author See **Owner** Wix **Status** Resolved

Schema IfcPropertyResource **Version** R1.5 - Pre-Beta

Issue Description Class: IfcCost -- How in the world can this be an Abstract class?

Proposed Solution Make it a concrete (instantiable) class.

Resolution Agreed.

Action # 1	***Assignee*** Wix	***Status*** Complete	***Resolved in Version*** R1.5 - Pre-Fin

JW will fix this. Confirmed in Pre-Final (RS).

Issue Number	***I - 071***	***Issue Date***	7/12/97

Author	See	***Owner***	Wix	***Status***	Resolved
Schema	IfcPropertyResource	***Version***	R1.5 - Pre-Beta		

Issue Description Class: IfcCost -- CostStage [STRING] - using a STRING is not very useful as we can expect that each application will use their own standard "Stages".

Proposed Solution Consider: Enumeration called IfcCostStageEnum which will allow multiple apps dealing with costs across stages to coordinate and support a common semantic meaning for each "Stage"

Resolution This cannot be well solved in R1.5. Remove CostStage from R1.5 and re-think a better way to handle this for R2.0.

(JW-980510) We have an R3 domain project ES-2 Cost Planning which is looking at the development of cost. We should either ask them to provide a definitive list of cost stages for use in R2 (I have done this) or wait until they complete their work for R3. When I get response from ES2, I will make the change.

Action # 1	***Assignee*** Wix	***Status*** Complete	***Resolved in Version*** R1.5 - Pre-Fin

1) JW - remove CostStage from R1.5. Confirmed in Pre-Final (RS).

Action # 2	***Assignee*** See	***Status*** Complete	***Resolved in Version*** R2.0 - Alpha

2) RS -add to list of R2.0 STF projects

Issue Number	***I - 072***	***Issue Date***	7/12/97

Author	See	***Owner***	Wix	***Status***	Resolved
Schema	IfcPropertyResource	***Version***	R1.5 - Pre-Beta		

Issue Description Class: IfcCost -- BasisNumber/BasisMeasure - These are only need for Unit Costs -- and therefore should be optional (two are mandatory now).

Proposed Solution Combine both 'BasisNumber' and 'BasisMeasure' into a single, optional attribute called "UnitCostBasis" of the type [IfcMeasureResource.IfcMeasureWithUnit] .

Resolution Agreed.

Action # 1	***Assignee*** Wix	***Status*** Complete	***Resolved in Version*** R1.5 - Pre-Fin

JW will make changes. Confirmed in Pre-Final (RS).

Issue Number	***I - 073***	***Issue Date***	7/12/97

Author	See	***Owner***	Wix	***Status***	Resolved
Schema	IfcPropertyResource	***Version***	R1.5 - Pre-Beta		

Issue Description Class: IfcCost -- BasisDate - this appears to be the date on which this cost was assigned, therefore it seems to be useful for ANY cost (not just Unit Costs).

Proposed Solution Change the 'BasisDate' to "CostDate" -- still optional.

Resolution Agreed.

Action # 1	***Assignee*** Wix	***Status*** Complete	***Resolved in Version*** R1.5 - Pre-Fin

JW to make the changes. Confirmed in Pre-Final (RS).

Issue Number	***I - 074***	***Issue Date***	7/12/97

Author	See	**Owner**	Wix	**Status**	Resolved
Schema	IfcPropertyResource	**Version**	R1.5 - Pre-Beta		

Issue Description Class: IfcCost --I don't see a way to reference a Bid (say from a contractor or sub-contractor). It seems like such cross referencing from summary/estimate cost items to component cost items (bids or estimates) will be important. Therefore, I would suggest considering the following:

Proposed Solution Add a mandatory attribute called "CostComponents L[0:N]". This will allow an estimator to roll up components (estimates or bids) into composite costs for assemblies -- directly in the cost model (as opposed to doing it only in a cost Schedule).

Resolution Agreed.

Action # 1 **Assignee** Wix **Status** Complete **Resolved in Version** R1.5 - Pre-Fin

JW to make the changes. Confirmed in Pre-Final (RS).

Issue Number I - 075 **Issue Date** 7/12/97

Author	See	**Owner**	Wix	**Status**	Rejected
Schema	IfcPropertyResource	**Version**	R1.5 - Pre-Beta		

Issue Description Superclass: IfcTypeDefResource.IfcPropertyDef -- Subtyping off of IfcPropertyDef is not shown in that schema.

Proposed Solution Update IfcTypeDefResource schema

Resolution Rejected -- this is a limitation of the tools we are using -- cannot show inheritence to another schema

---

Issue Number I - 076 **Issue Date** 7/12/97

Author	See	**Owner**	Wix	**Status**	Rejected
Schema	IfcPropertyResource	**Version**	R1.5 - Pre-Beta		

Issue Description Class: IfcClassification -- Table and Edition - given that the Notation (which is, of course, mandatory) is really dependent on the table and edition for a classification system, does it make sense for these to be optional?

Proposed Solution Consider: making Table and edition mandatory.

Resolution Rejected. This may reference an in-house classification system where there is not a table or edition.

Issue Number I - 077 **Issue Date** 7/12/97

Author	See	**Owner**	Liebich	**Status**	Resolved
Schema	IfcTypeDefResource	**Version**	R1.5 - Pre-Beta		

Issue Description Class: IfcPropertyTypeDef -- OwnerId, ProjectId, AuditTrail - These are the IfcRoot defined in the 'Pseudo Model' Therefore, this class should "have" an IfcRoot (using aggregation) -- see also issue I-3 in the review notes dated 8-Jul-97.

Proposed Solution Replace these three attributes with a mandatory attribute "PropertyIdAudit" of type IfcRoot (now shown as IfcKernelRoot in the Kernel Schema).

Resolution Resolved. See resolution described in I-42:
1) attach IfcSeed and IfcProjectUniqueID to IfcPropertyTypeDef
2) attach IfcProjectUniqueID to IfcPropertySet
3) contact Francois regarding why he argued for ProjectID on every atomic property (e.g. IfcSimpleProperty.

Tentative Design Policy decision: in order to lighten the identification load on the model, we need to identify the containers' that will have project unique ID and remove that ID from the contained objects -- e.g. no ID on each property, but only on the PropertySet which contains it -- AND -- no ID on geometry elements, but only on the ShapeRep in which the geometry is used.

Action # 1 **Assignee** Liebich **Status** Complete **Resolved in Version** R1.5 - Pre-Fin

TL to lead this study and work. Note: changes since this was captured. 1) IfcSeed is not IfcOwnerHistory. Confirmed with exception - IfcSeed refs in Kernel and IfcPropertyTypeResource should be updated to IfcOwnerHistory (RS email to TL - 15-Sep).

Issue Number I - 078 **Issue Date** 7/12/97

Author See **Owner** Liebich **Status** Resolved

Schema IfcTypeDefResource **Version** R1.5 - Pre-Beta

Issue Description Class: IfcPropertyTypeDef -- Type Driven Occurance Properties have been lost (see R1.0 spec). This is important because it is the type definition which identifies which of the "OccurrenceProperties" (on IfcObject) are associated with this Type. Without this reference, only the 'typing' application knows what was added into the "OccurrenceProperties". With this, any querying app can search and find the type driven OccurrenceProperties for this TypeDef. This will become imparative when we allow for object typing by different disciplines/domains/apps types.

Proposed Solution Add an attribute "OccurrencePropertySetName [STRING].

Resolution Alternative by TL is to add a reference from the Occurrence PropertySet to the TypeDef that drove it.

Action # 1 **Assignee** Liebich **Status** Complete **Resolved in Version** R1.5 - Pre-Fin

TL to study and fix. Confirmed in Pre-Final (RS). Note: further issue by RS on 2 added classes (IfcOccurrencePropertySet and IfcSharedPropertySet - subtyped from IfcPropertySet), just to allow this alternative (as opposed to method outlined above - used in R1.0).

Issue Number I - 079 **Issue Date** 7/12/97

Author See **Owner** Liebich **Status** Resolved

Schema IfcTypeDefResource **Version** R1.5 - Pre-Beta

Issue Description Class: IfcPropertyTypeDef -- We have already discussed at length the eventual need to be able to "Type" objects or "Groups" for multiple AEC industry perspectives (e.g. Architects view of a wall - Typed as exterior, interior, partition, etc. -- versus the structural engineer's view of a wall - Typed as bearing, shear, non-structural, etc.). We are VERY close to being able to do this now -- with two changes as recommended here and in the Kernel review of IfcObject.

Proposed Solution 1) add the attribute to IfcPropertyTypeDef -- "ObjTypeDomainView" which is an Enumeration [CrossDomain, Architect, HVAC, Structural, Civil, Constructor, FM],

2) on IfcObject -- change the optional 'TypeDefinedProperty' to a mandatory "TypeDefinitions" L[0:N] Ref [IfcTypeDefResource.IfcPropertyTypeDef].

This will allow multiple domain views to type the object (or Group) from their perspective. A list of TypeDefs (shared properties) will be referenced and a corresponding list of OccurrenceProperties will be attached.

Resolution Seems like and interesting idea, but should be double checked.

Agreed in email thread from 9/2-9/4 in order to support attachment of multiple type driven Occurrence PropertySets - defined on IfcObject.

Action # 1 **Assignee** Liebich **Status** Complete **Resolved in Version** R1.5 - Pre-Fin

TL to implement. Not confirmed (item 1 above not yet done) in Pre-Final (RS - email TL, 15-Sep).

Issue Number I - 080 **Issue Date** 7/12/97

Author See **Owner** Liebich **Status** Resolved

Schema IfcTypeDefResource **Version** R1.5 - Pre-Beta

Issue Description Class: IfcPropertyDef -- AttDescriptor - this should be optional since you have made this the Supertype for all pre-defined Properties as well as the Runtime defined ones. We don't need a descriptor for pre-defined simple attributes since each has a name and pre-defined semantic definition.

Proposed Solution Make this attribute optional.

Resolution TL solution is to move this to the two subtypes which need it (IFcPropertySet and IfcSimpleProperty) and thus remove it for the ones that don't need it.

Action #	1	***Assignee***	Liebich	***Status***	Complete	***Resolved in Version***	R1.5 - Pre-Fin

TL will fix it. Confirmed in Pre-Final (RS).

Issue Number	***I - 081***			***Issue Date***	8/8/97
Author	See	***Owner***	Liebich	***Status***	Resolved
Schema	IfcTypeDefResource	***Version***	R1.5 - Pre-Beta		

Issue Description Class: IfcPropertyDef -- OccurrenceReference [IfcPropertyDef] - this self reference is also confusing. The inverse relationship implies that the primary purpose for this is to allow occurrence level overriding of attributes -- either simple attributes or individual attributes within a Set. Is this right? Imagined usages cases --> 1) an application assiciates default values with a number of occurrences through the use of a type -- however, for special cases, the app can attach an individual property in the OccurrenceProperties list which provides an overriding value and points to the attribute in the share set which is superseded. Any receiving application must then replace the default value with the override. --- Is this correct?? If so, there is some question if you SHOULD allow this. The whole reason for standard types is to reduce construction costs through standardization and quantity pricing.

Proposed Solution Leave this overriding out unless application developers request it. Alternatively, let's do a member survey which asks if this should be allowed.

Resolution Agreed. This overriding will be removed for R1.5.

Action #	1	***Assignee***	Liebich	***Status***	Complete	***Resolved in Version***	R1.5 - Pre-Fin

TL to make the change. Confirmed in Pre-Final (RS).

Issue Number	***I - 082***			***Issue Date***	8/8/97
Author	See	***Owner***	Liebich	***Status***	Resolved
Schema	IfcKernel	***Version***	R1.5 - Pre-Beta		

Issue Description Class: IfcKernelRoot - Naming -- This should be designed to be used in the three places indicated in the ‘Meta Model’ developed on 30-May -- see also issue GI-3 above) AND should be named appropriately.

Proposed Solution Rename to IfcRoot

Resolution Resolved -- see resolution in I-42.

Action #	1	***Assignee***	Liebich	***Status***	Complete	***Resolved in Version***	R1.5 - Pre-Fin

TL will make the change. Confirmed in Pre-Final (RS).

Issue Number	***I - 083***			***Issue Date***	8/8/97
Author	See	***Owner***	Liebich	***Status***	Deferred to R2.0
Schema	IfcKernel	***Version***	R1.5 - Pre-Beta		

Issue Description Class: IfcKernelRoot - Attribute lost from R1.0 -- needs reference to IfcVersion (probably better named IfcObjectVersion).

Proposed Solution Create ObjectVersion object and add reference to it here.

Resolution Deferred to R2.0.

Action #	1	***Assignee***	See	***Status***	Complete	***Resolved in Version***	R2.0 - Alpha

RS to add to the list of STF projects for R2.0.

Issue Number	***I - 084***			***Issue Date***	8/8/97
Author	See	***Owner***	Liebich	***Status***	Resolved
Schema	IfcKernel	***Version***	R1.5 - Pre-Beta		

Issue Description Class: IfcProject - I would argue that projects are typed and may have associated properties just as the products they contain to. Additionally, projects in the firms I worked in were classified to support comparison and locating historical data in order to prepare proposals.

Proposed Solution 1) subtype from IfcObject
2) add a genericType
3) add classification.

Resolution Agreed - will be implemented as proposed.

Action #	1	***Assignee***	Liebich	***Status***	Complete	***Resolved in Version***	R1.5 - Pre-Fin

TL will make the change. Confirmed in Pre-Final (RS).

Issue Number	***I - 085***			***Issue Date***	8/8/97
Author	See	***Owner***	Liebich	***Status***	Resolved
Schema	IfcKernel	***Version***	R1.5 - Pre-Beta		

Issue Description Class: IfcObject - TypeDefinedProperty [IfcPropertyTypeDef] -- Naming issue, cardinality enhancement recommendation -- this is the TypeDefinition, which associates the shared properties and also drives the OccurrenceProperties. See also GI-6 (support for multiple TypeDefs from different domain points of view.

Proposed Solution Call it "TypeDefinition" and make it a mandatory attribute - List [0:N] --> this will also require the addition of a mandatory attribute "TypeForDomain" in the IfcPropertyTypeDef class -- the application defining type will have to define the Domain for which this ‘Type’ is valid/intended. We may also want to consider establishment of an enumeration of ‘standard’ domain/aplication view ‘Types’ so that we don’t end up with types defined for ‘Interior Designer’ and ‘Furniture Selection Rep’ when we want these two to be one.

Resolution Agreed to call it "TypeDefinition". See also I-79 regarding multiple TypeDefs and identification of the DomainViewType.

Action #	1	***Assignee***	Liebich	***Status***	Complete	***Resolved in Version***	R1.5 - Pre-Fin

TL to fix -- also pending results from checking with Implementers and Domain. Confirmed in Pre-Final with execption noted in I-79 (RS - email TL, 15-Sep).

Issue Number	***I - 086***			***Issue Date***	8/8/97
Author	See	***Owner***	Liebich	***Status***	Rejected
Schema	IfcKernel	***Version***	R1.5 - Pre-Beta		

Issue Description Class: IfcModelingAid - During the discussions in San Rafael late May (28/29/30), we re-introduced IfcControl as the supertype for ModelingAid and other types of constraints/controls.

Proposed Solution Remove IfcModelingAid from the Kernel and subtype from IfcControl in the IfcModelingAidExtension.

Resolution Rejected. Agreed that ModelingAid is not a control.

Issue Number	***I - 087***			***Issue Date***	8/8/97
Author	See	***Owner***	Liebich	***Status***	Resolved
Schema	IfcKernel	***Version***	R1.5 - Pre-Beta		

Issue Description Class: IfcConstructionAid - I don’t see a reason for including this in the models at all. It has no data and only a single relationship described on D2. Therefore, it has little or no semanic meaning and is not justified.

Proposed Solution Remove it from the R1.5 models and only re-introduce it when we have a definition and data which is specific enough to prevent mis-interpretaion.

Resolution Agreed -- May be reconsidered in R2.0.

Action #	1	***Assignee***	See	***Status***	Complete	***Resolved in Version***	R2.0 - Alpha

RS to add to the list of STF projects for R2.0.

Issue Number	***I - 088***			***Issue Date***	8/8/97

Author	See	**Owner**	Liebich	**Status**	Resolved
Schema	IfcKernel	**Version**	R1.5 - Pre-Beta		

Issue Description Class: IfcRelGroups - seems like we will need one or more attributes to assign a semantic meaning or purpose behind the grouping. This is one of the subtopics in XM-3 for R2.0.

Proposed Solution Add attribute "GroupPurpose [STRING]".

Resolution Agreed, except that it should be attached to the IfcGroup rather than IfcRelGroups.

Action # 1 **Assignee** Liebich **Status** Complete **Resolved in Version** R1.5 - Pre-Fin

TL to make the change. Confirmed in Pre-Final with exception that attribute is on IfcGroup rather than on the relationship (RS).

Issue Number I - 089 **Issue Date** 8/8/97

Author	See	**Owner**	Liebich	**Status**	Deferred to R2.0
Schema	IfcKernel	**Version**	R1.5 - Pre-Beta		

Issue Description Class: IfcRelUsesProducts and IfcRelUsesConstructionAids - These could be eliminated based on the typing of relationships proposed above (general issue GI-8 in the general comments for review 3C).

Proposed Solution Eliminate from the model.

Resolution Related to GI-8. Not resolved in first pass (21-Aug-97)

Second pass (23-Aug-97) - 11 classes (listed in GI-8) exist only to redeclare the RelatingObject and RelatedObjects. Still need to look for was to reduce this meaningless class count.

Action # 1 **Assignee** Liebich **Status** Incomplete **Resolved in Version** R2.0 - Alpha

TL/JW will look into a way of doing this with constraints in EXPRESS.

Action # 2 **Assignee** Wix **Status** Incomplete **Resolved in Version** R2.0 - Alpha

TL/JW will look into a way of doing this with constraints in EXPRESS.

Issue Number I - 090 **Issue Date** 8/8/97

Author	See	**Owner**	Liebich	**Status**	Deferred to R2.0
Schema	IfcKernel	**Version**	R1.5 - Pre-Beta		

Issue Description Class: IfcRelUsesProducts - This can be replaced by a ‘typed’ IfcRelationship1toN (see rationalization in GI-8 regarding typed relationships). Also, this is an awkward name.

Proposed Solution Replace with typed superclass.

Resolution Related to GI-8. Not resolved in first pass (21-Aug-97)

Second pass (23-Aug-97) - 11 classes (listed in GI-8) exist only to redeclare the RelatingObject and RelatedObjects. Still need to look for was to reduce this meaningless class count.

Action # 1 **Assignee** Liebich **Status** Incomplete **Resolved in Version** R2.0 - Alpha

TL/JW will look into a way of doing this with constraints in EXPRESS.

Action # 2 **Assignee** Wix **Status** Incomplete **Resolved in Version** R2.0 - Alpha

TL/JW will look into a way of doing this with constraints in EXPRESS.

Issue Number I - 091 **Issue Date** 8/8/97

Author	See	**Owner**	Liebich	**Status**	Resolved
Schema	IfcKernel	**Version**	R1.5 - Pre-Beta		

Issue Description Class: IfcProduct - Inverse relationships -- from IfcElement = HasReferencingElements (elements which declare they are related to this container) and HasElements (elements which declare they are owned by this container).

Proposed Solution Include these in the interface definition -- Note: we still need a way to include these in the EXG diagrams -- don't we ? Excluding them makes it difficult to understand the model from the EXG diagrams.

Resolution Agreed. These need to be included in the Interface definitions in the spreadsheet, but also in the documentation and EXPRESS code.

NOTE: this superseded by inclusion of generalized containment using IfcRelContains (subtype of IfcRelationship1toN). However, another issue is that there is no inverse relationship from IfcObject to IfcRelContains. This means that the only way to find out all the elements 'contained' in an object (say Building), is to iterate over the IfcRelContains rels and find the ones which reference the Building as the RelatingObject. --> logged as issue #313

Action # 1 **Assignee** See **Status** Eliminated **Resolved in Version** R1.5 - Final

RS to insure inclusion in the R1.5 SS. --

eliminated by the inclusion of generalized containment relationships.

Action # 2 **Assignee** Liebich **Status** Eliminated **Resolved in Version** R1.5 - Final

Include in EXPRESS code and documentation for IfcProduct. --

eliminated by the inclusion of generalized containment relationships.

Issue Number I - 092 **Issue Date** 8/8/97

Author See **Owner** Liebich **Status** Rejected

Schema IfcKernel **Version** R1.5 - Pre-Beta

Issue Description Class: IfcProduct - LocalPlacement [IfcLocalPlacement] -- this was the I_EntityPlacement in R1.0. Making it into an object -- seems okay. However, pushing it up to the IfcProduct class level creates an issue with respect to definition of a local placement for IfcNetwork, IfcSite, IfcSiteComplex, IfcBuildingComplex -- remember that we pushed this placement down to 4 places (from IfcProduct) just to avoid having placement on IfcSiteObject, IfcSiteComplex, IfcBuildingObject, IfcBuildingComplex . . .

Proposed Solution Consider: Personally, I like it this way because I have always argued that these containers should also have their own ShapeRep which is used in the early stages of design (before components have been designed) and in cases where abstract representation is needed. However, it does represent a shift is the consensus during the R1.0 discussions.

Resolution This is incorrect. These classes are subtyped from IfcGroup, which does not have LocalPlacement. Rejected.

Issue Number I - 093 **Issue Date** 8/8/97

Author See **Owner** Liebich **Status** Resolved

Schema IfcKernel **Version** R1.5 - Pre-Beta

Issue Description Class: IfcSequence - SequenceRelTo:IfcProcess, ResultsIn S[0:N] -- This appears to simply be a special case of a ‘Relationship1toN’ -- so it should not be subtyped from IfcRoot. Additionally, the "TimeLag" to successor processes may not always be the same. Consider: This could be a subtype of ‘Relationship1to1’ where there may be multiple IfcRelSequence objects associated with a process.

Proposed Solution Remove from the model as this can be a ‘typed’ ‘Relationship1to1’ or ‘Relationship1toN’ as described in general issue GI-8.

Resolution Related to GI-8. Not resolved in first pass (21-Aug-97)

Second pass (23-Aug-97) - IfcSequence is a 1toN relationship -- still need to solve the 'many to many' relationship problem on diagram 2. This will be revised to 1toN, Predecessor driven (e.g. RelatingObject = Predecessor, RelatedObject = Successors).

NOTE: this was superseded by I-200, in which IFcSequence was made a subtype of IfcRelationship1to1 instead.

Action # 1 **Assignee** Liebich **Status** Complete **Resolved in Version** R1.5 - Pre-Fin

TL will make changes. Confirmed in Pre-Final (RS).

Issue Number I - 094 **Issue Date** 8/8/97

Author See ***Owner*** Liebich ***Status*** Resolved

Schema IfcKernel ***Version*** R1.5 - Pre-Beta

Issue Description Class: IfcControl and IfcDocument - There is nothing defined for these classes. It appears that they are only included to provide structuring of the model. If so, they should be abstract.

Proposed Solution Make both abstract classes.

Resolution Agreed.

Action # 1 ***Assignee*** Liebich ***Status*** Complete ***Resolved in Version*** R1.5 - Pre-Fin

TL will make changes. Confirmed in Pre-Final (RS).

Issue Number ***I - 095*** ***Issue Date*** 8/8/97

Author See ***Owner*** Liebich ***Status*** Rejected

Schema IfcProductExt ***Version*** R1.5 - Pre-Beta

Issue Description General Issue for Schema - Building Element Containers still need their own geometry. See RS email on 970526 - "Re[2]: Open issues in ProductExt and SharedBldgElements".

Proposed Solution Add an optional "ContainerShape" to IfcGroup in the Kernel.

Resolution Rejected. The subtypes don't really need shape.

Issue Number ***I - 096*** ***Issue Date*** 8/8/97

Author See ***Owner*** Liebich ***Status*** Resolved

Schema IfcProductExt ***Version*** R1.5 - Pre-Beta

Issue Description Class: IfcSpatialElement (reference) - Error found -- called IfcSpatialObject in this reference.

Proposed Solution Correct to IfcSpatialElement.

Resolution Already resolved -- TL fixed it.

Issue Number ***I - 097*** ***Issue Date*** 8/8/97

Author See ***Owner*** Liebich ***Status*** Resolved

Schema IfcProductExt ***Version*** R1.5 - Pre-Beta

Issue Description Class: IfcSiteComplex, IfcBuildingComplex, IfcZone, IfcSystem - Ambiguous meaning for RelatedObjects allowed for each of these containers.

Proposed Solution Redeclare the specialized meanings for RelatedObjects for each of these containers -- see also GI-9.

Resolution Related to GI-9. Not resolved in first pass (21-Aug-97)

This must be handled in the documentation for R1.5.

Long term solution deferred to R2.0

Action # 1 ***Assignee*** See ***Status*** Complete ***Resolved in Version*** R2.0 - Alpha

Add research for long term solution to the list of projects for R2.0

Action # 2 ***Assignee*** Liebich ***Status*** Complete ***Resolved in Version*** R1.5 - Final

Enhance the reference documentation to clarify the meaning of RelatedObjects for these types.
WR added on IfcZone.

Issue Number ***I - 098*** ***Issue Date*** 8/8/97

Author See ***Owner*** Liebich ***Status*** Resolved

Schema IfcProductExt ***Version*** R1.5 - Pre-Beta

Issue Description Class: IfcSystem - Relationship lost from R1.0 -- In R1.0, we had a specialized relationship for IfcSystem --> IfcRelBldgSystems, which related a system to one or more buildings which it serviced.

Proposed Solution Add it back in.

Resolution Resolved. Add it back.

Action # 1 ***Assignee*** Liebich ***Status*** Complete ***Resolved in Version*** R1.5 - Pre-Fin

TL will make the change. Confirmed in Pre-Final (RS).

Issue Number ***I - 099*** ***Issue Date*** 8/8/97

Author See ***Owner*** Liebich ***Status*** Rejected

Schema IfcProductExt ***Version*** R1.5 - Pre-Beta

Issue Description Class: IfcRelConnectsElements & subtypes - Connections are Controls -- because they impose a geometric constraint on the connected elements. They are not Products.

Proposed Solution These entities should be moved to an IfcControls Schema. Note: the IfcControls schema is where I would anticipate we will put the general purpose constraint entities recommended by the Codes and Standards group.

Resolution Rejected. These are really Relationships, not Products or controls.

Issue Number ***I - 100*** ***Issue Date*** 8/8/97

Author See ***Owner*** Liebich ***Status*** Resolved

Schema IfcProductExt ***Version*** R1.5 - Pre-Beta

Issue Description Class: IfcRelConnectsElements & subtypes - These definitions are ambiguous and do not allow n-way connections. The subtypes define a single Point or Curve at which the connection is made, however they do not establish the corresponding point or curve within the "Connected" element’s geometry. Therefore, the "Connected" element(s) are floating with respect to the connection point/curve.

Proposed Solution 1) subtype from IfcRelationship1toN
2) define the "ConnectionPoint" and "ConnectionCurve" within the RelatingObject’s LCS
3) add the attributes "PointOnElements" and "CurveOnElements" to the two subtypes where these points/curves are defined in the LCS of the reference RelatedObjects.

Resolution TL agreed in principal, but not resolved in first pass (21-Aug-97). Compromise: Point currently defined in the ConnectionAtPoint relationship will be taken as being a point on the RelatingObject geometry (in its LCS). Another point will be added which is a point defined on the RelatedObject geometry (in its LCS). This second point will be optional. If the second point is omitted, the ReleatedObject will be connected at its origin (its placement location). Note: the compromise is that this is subtyped from IfcRelationship1to1, not 1toN.

Action # 1 ***Assignee*** See ***Status*** Complete ***Resolved in Version*** R1.5 - Pre-Fin

RS and TL will work on this process and make a proposal. Compromise: Point currently defined in the ConnectionAtPoint relationship will be taken as being a point on the RelatingObject geometry (in its LCS). Another point will be added which is a point defined on the RelatedObject geometry (in its LCS). This second point will be optional. If the second point is omitted, the ReleatedObject will be connected at its origin (its placement location). Note: the compromise is that this is subtyped from IfcRelationship1to1, not 1toN. Confirmed in Pre-Final (RS).

Action # 2 ***Assignee*** Liebich ***Status*** Complete ***Resolved in Version*** R1.5 - Pre-Fin

TL to implement agreed solution. Confirmed in Pre-Final (RS).

Issue Number ***I - 101*** ***Issue Date*** 8/8/97

Author See ***Owner*** Liebich ***Status*** Resolved

Schema IfcProductExt ***Version*** R1.5 - Pre-Beta

Issue Description Class: IfcRelConnectsElements & subtypes - Naming issues -- IfcRelConnectsByPoint, ByPoint, IfcRelConnectsByCurve and ByCurve are all a bit ‘forced’.

Proposed Solution Replace with IfcRelConnectedAtPoint, ConnectionPoint, IfcRelConnectedAtCurve, ConnectionCurve.

Resolution Agreed.

Action # 1 ***Assignee*** Liebich ***Status*** Complete ***Resolved in Version*** R1.5 - Pre-Fin

TL to make changes. Confirmed in Pre-Final (RS). Note: is actually IfcConnectsAtPoint.

Issue Number ***I - 102*** ***Issue Date*** 8/8/97

Author See ***Owner*** Liebich ***Status*** Deferred to R2.0

Schema IfcProductExt ***Version*** R1.5 - Pre-Beta

Issue Description Class: IfcElement - PerformedFunctions S[1:N] [IfcElementFunctionTypeEnum] -- I don't think that we are ready to introduce support for multi-funcitonality. This concept is CERTAINLY not well discussed or documented.

Proposed Solution Remove this concept until it has more discussion and explanation -- target for inclusion in R2.0.

Resolution Agreed (15-July-98)
NOTE: it has been proposed (by RJ/RS) that multi-functionality of elements is now handled in another way. See IfcElementGroupByFunction. Elements may belong to any number of functional groups. A number of The values that were in IfcSystemTypeEnum will be moved to the enum IfcFunctionTypeEnum because they were not systems, but were functional groups (e.g. Furnishings and SpaceSeparators).

Action # 1 ***Assignee*** Liebich ***Status*** Incomplete ***Resolved in Version*** R1.5 - Addend

Elminate PerformedFunctions from IfcElement and also IfcElementFunctionTypeEnum

Not complete as of 27-Nov-97 (RS) - overlap on "enclosure" for example - also, "furnishing", "Spacial" (note spelling error) and "Enclosing" are not systems.

Action # 2 ***Assignee*** Liebich ***Status*** Incomplete ***Resolved in Version*** R1.5 - Addend

Correct the IfcSystemTypeEnum to eliminate those that are not systems. Examples: "enclosure", "furnishing" and "Spatial"

Action # 3 ***Assignee*** Liebich ***Status*** Incomplete ***Resolved in Version*** R2.0 - Beta

TL/RS - Consider functional groups proposal for inclusion in R2.0

Action # 4 ***Assignee*** See ***Status*** Eliminated ***Resolved in Version*** R2.0 - Beta

TL/RS - Consider functional groups proposal for inclusion in R2.0

Issue Number ***I - 103*** ***Issue Date*** 8/8/97

Author See ***Owner*** Liebich ***Status*** Resolved

Schema IfcProductExt ***Version*** R1.5 - Pre-Beta

Issue Description Class: IfcElement - QuantityAccording [IfcMeasureResource.IfcUserDefinedType] -- after reading the documentation I would argue that this name is ambiguous. Also, the documentation states the data type as being a STRING.

Proposed Solution 1) rename to "QtyCalculationStd", 2) update the documentation to proper data type.

Resolution Agreed. The data type should be STRING. NOTE: this will be moved to an PropertySet in resolution to I-104.

Action # 1 ***Assignee*** Liebich ***Status*** Complete ***Resolved in Version*** R1.5 - Final

TL to make the changes in ProductExt. RS to make additions to PropertySets to be used with Elements (see action in I-104).

NOTE: this has been moved to a PropertySet along with the quantity attributes per the suggestion in I-104. Change to model confirmed, but not doc (RS).

Issue Number ***I - 104*** ***Issue Date*** 8/8/97

Author See ***Owner*** Liebich ***Status*** Resolved

Schema IfcProductExt ***Version*** R1.5 - Pre-Beta

Issue Description Class: IfcElement - calcQuantityByXxxx [various] -- This list of optional attributes is a bit tedious.

Proposed Solution Consider: these _could_ be defined as a standard PropertySet or as a List[0:N] IfcPropertyDef called calcQuantity.

Resolution Agreed. Move 5 quantities plus the QtyCalcStd attribute (see I-103) to an PropertySet called "Att_ElementQty".

Action # 1 **Assignee** Liebich **Status** Complete **Resolved in Version** R1.5 - Final

TL to remove attributes from IfcElement. Not confirmed in Pre-Final (RS - email TL, 15-Sep).

Action # 2 **Assignee** See **Status** Complete **Resolved in Version** R1.5 - Final

RS to create new PropertySet.

Issue Number I - 105 **Issue Date** 8/8/97

Author See **Owner** Liebich **Status** Resolved

Schema IfcProductExt **Version** R1.5 - Pre-Beta

Issue Description Class: IfcBuildingElement - This class appears to be included to provide model structure -- it appears that it should not be instantiated.

Proposed Solution Make it abstract.

Resolution Agreed.

Action # 1 **Assignee** Liebich **Status** Complete **Resolved in Version** R1.5 - Pre-Fin

TL to make the changes. Confirmed in Pre-Final (RS).

Issue Number I - 106 **Issue Date** 8/8/97

Author See **Owner** Liebich **Status** Resolved

Schema IfcProductExt **Version** R1.5 - Pre-Beta

Issue Description Class: IfcElementAssembly - This class appears to do the same thing as an IfcGroup, yet it is subtyped from IfcElement. If the relationship from IfcRelAssemblesElements were made to IfcBuildingElement, then BUILDING ELEMENTS COULD BE NESTED. This would be VERY powerful and desirable as elements could be approximations and illdefined in the early stages of design and more elaborate assemblies of component elements later in the design process. This parallels the design process and is VERY desirable.

Proposed Solution Eliminate IfcElementAssembly and redirect the relationship from IfcRelAssemblesElements to IfcBuilidngElement in order to allow any Building Element to be an assembly.

Resolution Agreed.

Action # 1 **Assignee** Liebich **Status** Complete **Resolved in Version** R1.5 - Pre-Fin

TL to make the changes. Confirmed in Pre-Final (RS) with exception that the relationships from IfcRelAssemblesElements are to the supertype, IfcElement - which also allows an 'assembly' (or grouping) of openings.

Issue Number I - 107 **Issue Date** 8/8/97

Author See **Owner** Liebich **Status** Resolved

Schema IfcProductExt **Version** R1.5 - Pre-Beta

Issue Description Class: IfcElementAssembly - IF NOT INTEGRATED INTO IFCBUILDINGELEMENT -- This class appears to be included to provide model structure -- it appears that it should not be instantiated.

Proposed Solution Make it abstract.

Resolution Already resolved. See resolution in I-106

Issue Number I - 108 **Issue Date** 8/8/97

Author See **Owner** Liebich **Status** Resolved

Schema IfcProductExt **Version** R1.5 - Pre-Beta

Issue Description Class: IfcBuilding - Buildings are definitely 'Typed' by Architects -- and I suspect they are by other disciplines as well.

Proposed Solution Add an optional attribute 'GeneicType [IfcBldgTypeEnum]. Also define the enumeration and associated PropertySets.

Resolution Agreed.

Action #	1	***Assignee***	Liebich	***Status***	Complete	***Resolved in Version***	R1.5 - Pre-Fin

TL to make the changes. Confirmed in Pre-Final (RS).

Issue Number I - 109 ***Issue Date*** 8/8/97

Author	See	***Owner***	Liebich	***Status***	Rejected
Schema	IfcProductExt	***Version***	R1.5 - Pre-Beta		

Issue Description Class: IfcBuilding - calcTotalHeight, calcSiteCoverage, calcTotalVolume -- This list of optional attributes is a bit tedious.

Proposed Solution Consider: these could be defined as a standard PropertySet or as a List[0:N] IfcPropertyDef called calcBldgQuantity.

Resolution Reject. Not agreed. These are semantically specific to these classes (and not a bunch of subtypes). Therefore, they should stay.

Issue Number I - 110 ***Issue Date*** 8/8/97

Author	See	***Owner***	Liebich	***Status***	Rejected
Schema	IfcProductExt	***Version***	R1.5 - Pre-Beta		

Issue Description Class: IfcBuilding - Redeclaration of containment relationship with IfcBuildingComplex.

Proposed Solution Redeclare relationships from IfcRelBldgsComplex -- RelatingObject = IfcBldgComplex, RelatedObjects = IfcBuilding.

Resolution Rejected. This was in R1.0. It has been replaced by the general purpose grouping mechanism in R1.5.

Issue Number I - 111 ***Issue Date*** 8/8/97

Author	See	***Owner***	Liebich	***Status***	Resolved
Schema	IfcProductExt	***Version***	R1.5 - Pre-Beta		

Issue Description Class: IfcBuilding - R1.0 Objectified relationship "IfcRelBldgService" has disappeared -- Redeclaration of the Relationship1toN needed?

Proposed Solution Add IfcRelBldgService where -- RelatingObject = IfcBuilding, RelatedObjects = IfcSystem.

Resolution Agreed.

Action #	1	***Assignee***	Liebich	***Status***	Complete	***Resolved in Version***	R1.5 - Pre-Fin

TL to make the changes. Confirmed in Pre-Final (RS), with exception that the relationship is reversed -- that is, a System may service multiple Buildings.

Issue Number I - 112 ***Issue Date*** 8/8/97

Author	See	***Owner***	Liebich	***Status***	Resolved
Schema	IfcProductExt	***Version***	R1.5 - Pre-Beta		

Issue Description Class: IfcBuildingStorey - There are definitely cases where it would be useful to allow 'Typing' of BuildingStoreys (e.g. Retail, Business Offices, Mechanical Equipment, Interstitial).

Proposed Solution Add an optional attribute 'GeneicType [IfcBldgTypeEnum]. Also define the enumeration and associated PropertySets.

Resolution Resolved. Reference solution in I-108.

Action #	1	***Assignee***	Liebich	***Status***	Complete	***Resolved in Version***	R1.5 - Pre-Fin

TL to make the changes. Confirmed in Pre-Final (RS).

Issue Number ***I - 113*** ***Issue Date*** 8/8/97

Author See ***Owner*** Liebich ***Status*** Rejected

Schema IfcProductExt ***Version*** R1.5 - Pre-Beta

Issue Description Class: IfcBuildingStorey - calcTotalHeight, calcTotalArea, calcTotalVolume -- This list of optional attributes is a bit tedious.

Proposed Solution Consider: these could be defined as a standard PropertySet or as a List[0:N] IfcPropertyDef called calcBldgStoreyQuantity.

Resolution Rejected. Not agreed. These are semantically specific to these classes (and not a bunch of subtypes). Therefore, they should stay.

Issue Number ***I - 114*** ***Issue Date*** 8/8/97

Author See ***Owner*** Liebich ***Status*** Resolved

Schema IfcProductExt ***Version*** R1.5 - Pre-Beta

Issue Description Class: IfcBuildingStorey - PartOfBuilding [IfcBuilding] -- this containment relationship is declared explicitly where such relationships are handled by the general purpose '1toN' relationship mechanism in almost all other cases.

Proposed Solution Consider: does this make it redundant? Is there a problem?

Resolution For the sake of consistency, create an objectified relationship between Building and BuildingStorey.

Action # 1 ***Assignee*** Liebich ***Status*** Complete ***Resolved in Version*** R1.5 - Pre-Fin

TL to make the changes. Confirmed in Pre-Final (RS), with execption that this is not an explicit objectified relationship, it is one of may 'uses' of the IfcRelContains, defined in the Kernel.

Issue Number ***I - 115*** ***Issue Date*** 8/8/97

Author See ***Owner*** Liebich ***Status*** Resolved

Schema IfcProductExt ***Version*** R1.5 - Pre-Beta

Issue Description Class: IfcBuildingSection, IfcBuildingSubStorey - Currently these classes have nothing defined in them -- therefore the need for them is questionable. However, I could see the case for justifying them on the basis that they could be typed -- e.g. Entry Foyer, Stair Tower, Core, Manufacturing Wing, etc.

Proposed Solution If we are to keep these two, They should include attributes "GenericType" and data types [BldgSectionTypeEnum] and [BldgSubStoreyTypeEnum].

Resolution IF Building Section and BuildingSubStorey are kept in the model (JW checking with Steve Race for his input on this) --> then agreed.

Action # 1 ***Assignee*** Liebich ***Status*** Eliminated ***Resolved in Version*** R1.5 - Pre-Fin

TL to make the changes. BuildingSubStorey eliminated, but BuildingSection kept. The change NOT confirmed in Pre-Final (RS email to TL, 15-Sep). This is not possible in EXPRESS since BuildingSection is subtyped from Building - which already has "GenericType". Action eliminated.

Issue Number ***I - 116*** ***Issue Date*** 8/8/97

Author See ***Owner*** Liebich ***Status*** Resolved

Schema IfcProductExt ***Version*** R1.5 - Pre-Beta

Issue Description Class: IfcBuildingSection, IfcBuildingSubStorey - PartOfBuilding [IfcBuilding], PartOfStorey [IfcBuildingStorey] -- these containment relationships are declared explicitly where such relationships are handled by the general purpose '1toN' relationship mechanism in almost all other cases. Additionally, this appears to be redundant with the "ReferencesContainers" and "PartOfContainer" attributes on the IfcElement supertype.

Proposed Solution Consider: does this make it redundant? Is there a problem?

Resolution For the sake of consistency, create an objectified relationship between Building and BuildingSection. IfcSubStorey is now gone. See I-192.

Action # 1 **Assignee** Liebich **Status** Complete **Resolved in Version** R1.5 - Pre-Fin

TL to make the changes. Confirmed in Pre-Final (RS), with execption that this is not an explicit objectified relationship, it is one of may 'uses' of the IfcRelContains, defined in the Kernel.

Issue Number ***I - 117*** **Issue Date** 8/8/97

Author See **Owner** Liebich **Status** Rejected

Schema IfcProductExt **Version** R1.5 - Pre-Beta

Issue Description Class: IfcSpaceBoundary - PhysicalOrVirtual [BOOLEAN] -- This attribute appears to be redundant. The answer to the question can be derived from the INV relationship to IfcRelSeparatesSpaces.RelatedObjects L[1:N]. If this INV relationship is not NULL, then there is one or more physical elements creating the boundary -- therefore it will be "Physical". Conversely, if the relationship is NULL, then the boundary must be "Virtual"

Proposed Solution Remove the attribute.

Resolution Rejected. This can be used to communicate design intent -- this boundary SHOULD be virtual.

Issue Number ***I - 118*** **Issue Date** 8/8/97

Author See **Owner** Liebich **Status** Rejected

Schema IfcProductExt **Version** R1.5 - Pre-Beta

Issue Description Class: IfcSpace - calcTotalPerimeter, calcTotalArea, calcTotalVolume -- This list of optional attributes is a bit tedious.

Proposed Solution Consider: these could be defined as a standard PropertySet or as a List[0:N] IfcPropertyDef called calcSpaceQuantity.

Resolution Reject. Not agreed. These are semantically specific to these classes (and not a bunch of subtypes). Therefore, they should stay.

Issue Number ***I - 119*** **Issue Date** 8/8/97

Author See **Owner** Liebich **Status** Resolved

Schema IfcProductExt **Version** R1.5 - Pre-Beta

Issue Description Class: IfcSpace - Access to contained elements is rather inconvenient now. Where in R1.0 we had an attribute "HasElements" which gave us direct access, we now have only indirect access through the INV relationship - IfcProduct.HasElements S[0:N].

Proposed Solution Insure that these inverse relationships are exposed through interfaces in the IDL model view.

Resolution Agreed. -- However, this is even more different after introduction of generalized containment relationships (IfcProduct.HasElements is now missing too!). Resolution actions eliminated and problem restated for current model in I-313.

Action # 1 **Assignee** See **Status** Eliminated **Resolved in Version** R1.5 - Final

RS to insure that this is exposed in the SS and thus the IDL interfaces.

Eliminated because this is now invalide due to introduction of generalized containment relationships. See actions from I-313 --> which restates problem for resulting model configuration.

Issue Number ***I - 120*** **Issue Date** 8/8/97

Author See **Owner** Liebich **Status** Resolved

Schema IfcProductExt **Version** R1.5 - Pre-Beta

Issue Description Class: IfcRelSeparatesSpaces - RelatingObject [IfcElement] -- It appears to me that this data type should really be IfcBuildingElement (so long as IfcBuildingElement and IfcElementAssembly are combined as recommended).

Proposed Solution Change to IfcBuildingElement.

Resolution Agreed.

Action # 1 ***Assignee*** Liebich ***Status*** Complete ***Resolved in Version*** R1.5 - Pre-Fin

TL to make the change. Confirmed in Pre-Final (RS).

Issue Number ***I - 121*** ***Issue Date*** 8/8/97

Author See ***Owner*** Liebich ***Status*** Rejected

Schema IfcProductExt ***Version*** R1.5 - Pre-Beta

Issue Description Class: IfcRelSeparatesSpaces - RelatingObject, RelatedObjects L[1:N] -- it appears that the direction of these is reversed from what would be normal -- that is, a SpaceBoundary would normally be defined by one or more Elements --> therefore, the RelatingObject should be the IfcSpaceBoundary and the RelatedObjects should be the IfcBuildingElements.

Proposed Solution Reverse the directions and cardinality of these relationships.

Resolution Rejected. The SpaceBoundary should be broken up so that there is never more than one BuildingElement per SpaceBoundary.

Issue Number ***I - 122*** ***Issue Date*** 8/8/97

Author See ***Owner*** Liebich ***Status*** Resolved

Schema IfcProductExt ***Version*** R1.5 - Pre-Beta

Issue Description Class: IfcPartialSpace - why create another subtyped class when the same thing could be accomplished by simply allowing Spaces to contain Spaces -- something which does not appear to be prevented in any event!

Proposed Solution Allow Spaces to be nested (to contain other spaces) and eliminate this class.

Resolution Will eliminate IfcPartialSpace and allow nesting of Spaces in the same way as for IfcBuildingElement (see I-106) using an objectified relationship.

Action # 1 ***Assignee*** Liebich ***Status*** Complete ***Resolved in Version*** R1.5 - Pre-Fin

TL to make changes. Confirmed in Pre-Final (RS).

Issue Number ***I - 123*** ***Issue Date*** 8/8/97

Author See ***Owner*** Liebich ***Status*** Rejected

Schema IfcProductExt ***Version*** R1.5 - Pre-Beta

Issue Description Class: IfcPartialSpace - IF THIS CLASS IS KEPT -- one rationalization would be to provide a Domain or Functional Point of View (POV). In most cases, partial spaces are defined from the point of view of a particular domain or application.

Proposed Solution Add an optional attribute "FunctionalPOV [IfcFuncPovTypeEnum]".   Then define the enumeration. I believe this ties in with the explanation in the .DOC file.

Resolution Rejected. Will not be keeping Partial Space.

Issue Number ***I - 124*** ***Issue Date*** 8/8/97

Author See ***Owner*** Liebich ***Status*** Resolved

Schema IfcProductExt ***Version*** R1.5 - Pre-Beta

Issue Description Class: IfcSite - SiteGeometry (Contours and boundaries) AND BuildableVolumeGeometry -- which were defined in R1.0 -- are missing. This is a BIG problem as these information sets are VERY commonly used by the project team

Proposed Solution Add these back in.

Resolution Agreed.

Action # 1 ***Assignee*** Liebich ***Status*** Complete ***Resolved in Version*** R1.5 - Final

TL to add. See also I-194.

Diagrams added, IfcShapeRepresentation.UsageTag used to distinguish.

Issue Number	***I - 125***			***Issue Date***	8/8/97
Author	See	***Owner***	Liebich	***Status***	Resolved
Schema	IfcProductExt	***Version***	R1.5 - Pre-Beta		

Issue Description Class: IfcSite - calcTotalPerimeter (not yet defined), calcSiteArea, calcBuildableVolume (not yet defined) -- This list of optional attributes could be handled in the same way proposed for Building, BuildingStorey and Space.

Proposed Solution Consider: these could be defined as a standard PropertySet or as a List[0:N] IfcPropertyDef called calcSiteQuantity.

Resolution Agree to add "calcTotalPerimeter" but not to make the ParameterSet.

Action # 1 ***Assignee*** Liebich ***Status*** Complete ***Resolved in Version*** R1.5 - Pre-Fin

TL to make changes. Note confirmed in Pre-Final (RS email to TL, 15-Sep).

Issue Number	***I - 126***			***Issue Date***	8/8/97
Author	See	***Owner***	Liebich	***Status***	Resolved
Schema	IfcProductExt	***Version***	R1.5 - Pre-Beta		

Issue Description Class: IfcSite - Redeclaration of containment relationship with IfcSiteComplex.

Proposed Solution Redeclare relationships from IfcRelSiteComplex -- RelatingObject = IfcSiteComplex, RelatedObjects = IfcSite.

Resolution IfcSiteComplex was eliminated -- we are using IfcGroup with "Purpose" = "Site Complex".

Issue Number	***I - 127***			***Issue Date***	8/8/97
Author	See	***Owner***	Liebich	***Status***	Resolved
Schema	IfcProductExt	***Version***	R1.5 - Pre-Beta		

Issue Description Class: IfcSite - Access to contained elements is rather inconvenient now. Where in R1.0 we had attributes "HasBuildings" and "HasElements", which gave us direct access, we now have only indirect access through the INV relationships - IfcProduct.HasElements S[0:N] and IfcSite.HasBuildings

Proposed Solution Insure that these inverse relationships are exposed through interfaces in the IDL model view.

Resolution Agreed. -- -- However, this is even more different after introduction of generalized containment relationships
(IfcProduct.HasElements S[0:N] and IfcSite.HasBuildings are now missing too!).

Resolution actions eliminated and problem restated for current model in I-313.

Action # 1 ***Assignee*** See ***Status*** Eliminated ***Resolved in Version*** R1.5 - Final

RS to insure this is included in the SS and JL to include in the IDL.

Eliminated because this is now invalide due to introduction of generalized containment relationships. See actions from I-313 --> which restates problem for resulting model configuration.

Issue Number	***I - 128***			***Issue Date***	8/8/97
Author	See	***Owner***	Wix	***Status***	Resolved
Schema	IfcProcessExt	***Version***	R1.5 - Pre-Beta		

Issue Description Class: IfcRelGroupsWorks - Naming issue -- Groups Works sounds clumsey .

Proposed Solution "IfcRelGroupsWork" (drop the plural on work) OR "IfcGroupsWorkTasks".

Resolution This subtype of IfcRelGroups was eliminated in favor of using the generalized IfcRelGroups. The resulting IfcGroup.Purpose = "Groups Work Tasks".

Action # 1 ***Assignee*** Wix ***Status*** Eliminated ***Resolved in Version*** R1.5 - Pre-Fin

JW will make the change. Confirmed in Pre-Final (RS)

Issue Number I - 129 | **Issue Date** 8/8/97

Author See | **Owner** Wix | **Status** Resolved

Schema IfcProcessExt | **Version** R1.5 - Pre-Beta

Issue Description Class: IfcRelGroupsWorks - redeclaration of the RelatedObjects side of the relationship is missing.

Proposed Solution Add redeclared SELF\IfcRelationship1toN.RelatedObjects L[1:N] [IfcWorkTask]. It would also be useful to rename this relationship to "HasWorkTasks" and the INV "PartOfWorkGroup" (see also GI-10).

Resolution Obsolete. Since this attribute no longer exists (see I-128), we don't need to rename.

Action # 1 | **Assignee** Wix | **Status** Eliminated | **Resolved in Version** R1.5 - Pre-Fin

JW will make the change. Not confirmed in Pre-Final (RS email to JW, 15-Sep).

Issue Number I - 130 | **Issue Date** 8/8/97

Author See | **Owner** Wix | **Status** Resolved

Schema IfcProcessExt | **Version** R1.5 - Pre-Beta

Issue Description Class: IfcWorkGroup - WorkSectionID [STRING], WorkSectionName [STRING] -- naming issue -- these must carry over from an old naming of WorkSection.

Proposed Solution Rename to "IfcGroupID" and "IfcGroupName".

Resolution Agreed.

Action # 1 | **Assignee** Wix | **Status** Complete | **Resolved in Version** R1.5 - Pre-Fin

JW will make the change. Confirmed in Pre-Final (RS)

Issue Number I - 131 | **Issue Date** 8/8/97

Author See | **Owner** Wix | **Status** Resolved

Schema IfcProcessExt | **Version** R1.5 - Pre-Beta

Issue Description Class: IfcWorkTask - It appears that we have lost one of the most important things we had in the R1.0 Process Model (as argued by the Estimating and Construction guys in the NA) --> I_ResourceUse -- which included Resources, ResourceQuantity and ResourceDuration.

Proposed Solution 1) create a new object called IfcResourceUse which includes these three things defined in the I_ResourceUse interface on IfcWorkTask from R1.0. 2) add an attribute on IfcWorkTask --> ResourceUse L[1:N] [IfcResourceUse]

Resolution Agreed:
1) Add IfcResource at the Kernel level
2) Add the IfcResoureUse class as described above.
3) Add attribute "ResourceUse" as described above.

Action # 1 | **Assignee** Liebich | **Status** Complete | **Resolved in Version** R1.5 - Pre-Fin

TL to make change to Kernel. JW to make changes to Process model. Confirmed in Pre-Final (RS)

Action # 2 | **Assignee** Wix | **Status** Complete | **Resolved in Version** R1.5 - Pre-Fin

TL to make change to Kernel. JW to make changes to Process model. Confirmed in Pre-Final (RS)

Issue Number I - 132 | **Issue Date** 8/8/97

Author See | **Owner** Wix | **Status** Resolved

Schema IfcProcessExt | **Version** R1.5 - Pre-Beta

Issue Description Class: IfcWorkTask - TaskCost [IfcPropertyRes.IfcCost] and WorkMethod [STRING], both of which were defined in R1.0 are missing.

Proposed Solution Add them back in -- both optional.

Resolution Agreed.

Action # 1 ***Assignee*** Wix ***Status*** Complete ***Resolved in Version*** R1.5 - Pre-Fin

JW to make the change. Confirmed in Pre-Final (RS).

Issue Number ***I - 133*** ***Issue Date*** 8/8/97

Author See ***Owner*** Wix ***Status*** Resolved

Schema IfcProcessExt ***Version*** R1.5 - Pre-Beta

Issue Description Class: IfcWorkTask - TaskNumberID [STRING] and WorkSchedule [IfcWorkSchedule] -- both of these attribute names are not very semantically accurate.

Proposed Solution change them to "WorkTaskID [STRING]" and "WorkTaskSchedule [IfcWorkTaskSchedule] ".

Resolution Agreed.

Action # 1 ***Assignee*** Wix ***Status*** Complete ***Resolved in Version*** R1.5 - Pre-Fin

JW to make the change. Not completely confirmed in Pre-Final (RS - email JW 15-Sep) - TaskNumberID not yet renamed.

Issue Number ***I - 134*** ***Issue Date*** 8/8/97

Author See ***Owner*** Wix ***Status*** Resolved

Schema IfcProcessExt ***Version*** R1.5 - Pre-Beta

Issue Description Class: IfcWorkSchedule - Classname is not semantically accurate.

Proposed Solution Rename it to "IfcWorkTaskSchedule".

Resolution Agreed.

Action # 1 ***Assignee*** Wix ***Status*** Complete ***Resolved in Version*** R1.5 - Pre-Fin

JW to make the change. Confirmed in Pre-Final (RS).

Issue Number ***I - 135*** ***Issue Date*** 8/8/97

Author See ***Owner*** Wix ***Status*** Rejected

Schema IfcProcessExt ***Version*** R1.5 - Pre-Beta

Issue Description Class: IfcWorkSchedule - The only attribute shown as mandatory on this class is ID. Surely Status, Duration, ScheduledStart should be mandatory also ??

Proposed Solution Change Status, Duration, ScheduledStart to mandatory.

Resolution Rejected. A schedule object may be created before you know the start date or duration. Then information filled in over time.

Issue Number ***I - 136*** ***Issue Date*** 8/8/97

Author See ***Owner*** Wix ***Status*** Resolved

Schema IfcProcessExt ***Version*** R1.5 - Pre-Beta

Issue Description Class: IfcWorkSchedule - ScheduleDuration [IfcMeasureResource.IfcTimeDuration] -- At first I was confused as to whether this was the duration between the "early" dates, the "late" dates, or the "scheduled" dates. The documentation does say duration "scheduled", but it can be confusing.

Proposed Solution Change the name of the attribute to "ScheduledDuration" (note the "d").

Resolution Agreed.

Action # 1 ***Assignee*** Wix ***Status*** Complete ***Resolved in Version*** R1.5 - Pre-Fin

JW to make the change. Confirmed in Pre-Final (RS).

Issue Number ***I - 137*** ***Issue Date*** 8/8/97

Author See ***Owner*** See ***Status*** Rejected

Schema	IfcModelingAidExt	***Version***	R1.5 - Pre-Beta

Issue Description Class: IfcModelingAid - Model Structure -- as discussed in the issues for the Kernel, It was my understanding that we agreed in late May that IfcmodelingAid should be subtyped from IfcControl. If that is the case, it should not be defined in the Kernel, but as a subtype of IfcControl in this schema.

Proposed Solution Move IfcModelingAid class to this schema (from Kernel) and subtype from IfcKernel.IfcControl.

Resolution Rejected. It was agreed that a ModelingAid is not a Control.

Issue Number ***I - 138*** ***Issue Date*** 8/8/97

Author	See	***Owner***	See	***Status***	Resolved
Schema	IfcModelingAidExt	***Version***	R1.5 - Pre-Beta		

Issue Description Class: Proposed new classes - IfcRefPoint, IfcRefCurve, IfcRefFace -- these utility classes will be used as references in the placement of other elements. The reason we need them (rather than using the geometry entities directly) is that our LocalPlacement relates the Axis2Placement to an IfcObject. This means that the geometry entities cannot be used directly, but must be wrapped and used as ModelingAids. The first and most common practical application of these is in the definition of Reference lines for the placement of Walls (ref. the discussions with our Japanese chapter developers).

Proposed Solution Create 3 new classes -- subtyped from IfcModelingAid --> IfcRefPoint (which has a relationship named "RefPoint" to [IfcGeometry.IfcCartesianPoint]), IfcRefCurve(which has a relationship named "RefCurve" to [IfcGeometry.IfcCurve]), IfcRefFace (which has a relationship named "RefSurface" to [IfcGeometry.IfcSurface]).

Resolution Not resolved in first pass (21-Aug-97). Second Pass (23-Aug-97) - Agreed.

Action # 1 ***Assignee*** See ***Status*** Complete ***Resolved in Version*** R1.5 - Final

RS will make changes. Confirmed in Pre-Final (RS).

Issue Number ***I - 139*** ***Issue Date*** 8/8/97

Author	See	***Owner***	See	***Status***	Resolved
Schema	IfcModelingAidExt	***Version***	R1.5 - Pre-Beta		

Issue Description Class: IfcPlacementRelToGrid - I am not really convinced that we need this special type of placement. I don't find the added attributes (OffsetToGridAxis, DistanceToCrossingAxes, CrossingNearIntersection) to be particularly useful to applications -- although I do acknowledge that some of the attributes defined in some of the Architecture group's attribute sets could make use of some of these.

Proposed Solution Consider using the default LocalPlacements or come up with strong rationalizations for the value in the added attributes.

Resolution Not resolved in first pass (21-Aug-97). Second pass (23-Aug-97) - Proposal for a more generalize solution for "Constrained" Placements was discussed and will be finalized by TL and RS. NOTE: this may mean that the ModelingAids cannot be moved down to the Resource Layer.

Action # 1 ***Assignee*** Liebich ***Status*** Complete ***Resolved in Version*** R1.5 - Pre-Fin

TL and RS to finalize for inclusion in Kernel. IfcConstrainedPlacement (relative to Curves) will now be defined in IfcModelingAid schema. IfcLocalPlacement was also moved into this schema. See notes from 7-Sep-97 mtg. Confirmed in Pre-Final (RS) - IfcConstrainedPlacement subtyped from IfcLocalPlacement - and allows constraint of one or both end points of a path - using an IfcPlacementConstraint, the first subtype of which is IfcConstraintRelIntersection.

Action # 2 ***Assignee*** See ***Status*** Complete ***Resolved in Version*** R1.5 - Pre-Fin

TL and RS to finalize for inclusion in Kernel. IfcConstrainedPlacement (relative to Curves) will now be defined in IfcModelingAid schema. IfcLocalPlacement was also moved into this schema. See notes from 7-Sep-97 mtg. Confirmed in Pre-Final (RS) - IfcConstrainedPlacement subtyped from IfcLocalPlacement - and allows constraint of one or both end points of a path - using an IfcPlacementConstraint, the first subtype of which is IfcConstraintRelIntersection.

Issue Number ***I - 140*** ***Issue Date*** 8/8/97

Author	See	***Owner***	See	***Status***	Resolved
Schema	IfcModelingAidExt	***Version***	R1.5 - Pre-Beta		

Issue Description Class: IfcPlacementRelToGrid - SELF\IfcPlacement.PlacementRelTo [IfcGridAxis] -- Placement relative to grids is QUITE OFTEN relative to intersections, not just axes.

Proposed Solution Generalize this to be relative to any of the grid related object types.

Resolution Not resolved in first pass (21-Aug-97). Second Pass (23-Aug-97) - resolved - see I-139

Issue Number I - 141 ***Issue Date*** 8/8/97

Author	See	***Owner***	Wix	***Status***	Resolved
Schema	IfcDocumentExt	***Version***	R1.5 - Pre-Beta		

Issue Description General issues for schema - Schema content - I am a bit troubled by the fact that the cost schedule and general purpose table that are the only contents of this schema are not really documents. They are general purpose data structures that may be presented (or partially presented) in documents. I have been viewing the DocumentsExtension as the place where we build links to and from real documents, but that we stop short of trying to capture the actual content of these documents (or else we will be trying to model the whole world).

Proposed Solution Consider: Since they are general purpose, maybe a better location for these would be at the Resource Layer. I believe this is particularly true for the general purpose table; although, since cost is such an important factor in all decisions, I would make the case for the CostSchedule (CostEstimate) as well.

Complication: Since the CostSchedule schema uses an objectified relationship, it would be difficult to push it to the resource layer without also moving the root for objectified relationships to that layer as well.

Resolution Partial agreement. Push general purpose tables to the Resource Layer and create a new Resource called "IfcUtilityResources". Leave CostSchedule as it is.

Action # 1 ***Assignee*** Wix ***Status*** Complete ***Resolved in Version*** R1.5 - Pre-Fin

JW to make changes. Confirmed in Pre-Final (RS).

Issue Number I - 142 ***Issue Date*** 8/8/97

Author	See	***Owner***	Wix	***Status***	Deferred to R2.0
Schema	IfcDocumentExt	***Version***	R1.5 - Pre-Beta		

Issue Description General schema issues - Sub-Schema naming -- In looking at the entities included in the Cost Schedule, I would argue that this is not really a Cost Schedule, but the data structures for a Cost Estimate instead. A schedule includes provisions for presentation in a document -- this does not. Having said this, I think that it is EVEN MORE USEFUL to include a Cost Estimate schema because it is more general purpose than a Cost Schedule.

Proposed Solution 1) Change the name of this sub-schema to CostEstimate. 2) change the names of the following 4 entities: IfcCostSchedule ? IfcCostEstimate, IfcCostScheduleGroup ? IfcCostEstimateGroup, IfcCostScheduleElement ? IfcCostEstimateElement, IfcRelGroupsCostSchedules ? IfcRelGroupsCostEstimate

Resolution Deferred until R2.0

Action # 1 ***Assignee*** See ***Status*** Complete ***Resolved in Version*** R2.0 - Alpha

RS to add to the list of STF projects for R2.0.

Issue Number I - 143 ***Issue Date*** 8/8/97

Author	See	***Owner***	Wix	***Status***	Resolved
Schema	IfcDocumentExt	***Version***	R1.5 - Pre-Beta		

Issue Description Class: IfcRelGroupsCostSchedules - RelatingObject [IfcCostScheduleGroup], RelatedObjects L[1:N] [IfcCostScheduleOrGroup] -- The direction of these relationships is backwards -- that is, a Schedule includes one or more other schedules or groups, which may include other schedules or groups, etc.

Proposed Solution Reverse the 'Relating' and 'Related' directions and cardinality.

Resolution IfcRelGroupsCostSchedules eliminated. IfcRelGroups used instead. Resulting IfcGroup.Purpose = "Groups Cost Schedules".

No action required as this issue was eliminated.

Action # 1 **Assignee** Liebich **Status** Eliminated **Resolved in Version** R1.5 - Pre-Fin

TL/JW to make the changes. Not confirmed in Pre-Final (RS email JW, 15-Sep).

Action # 2 **Assignee** Wix **Status** Eliminated **Resolved in Version** R1.5 - Pre-Fin

TL/JW to make the changes. Not confirmed in Pre-Final (RS email JW, 15-Sep).

Issue Number ***I - 144*** **Issue Date** 8/8/97

Author See **Owner** Wix **Status** Resolved

Schema IfcDocumentExt **Version** R1.5 - Pre-Beta

Issue Description Class: IfcCostScheduleGroup - Attribute missing (?) -- the attribute "GroupNumber" was included in R1.0 but is missing now.

Proposed Solution Add it back in as "GroupIdentifier [STRING]"

Resolution Agreed --> "GroupID"

Action # 1 **Assignee** Wix **Status** Complete **Resolved in Version** R1.5 - Pre-Fin

JW to make the changes. Confirmed in Pre-Final (RS) -- named "GroupID".

Issue Number ***I - 145*** **Issue Date** 8/8/97

Author See **Owner** Wix **Status** Resolved

Schema IfcDocumentExt **Version** R1.5 - Pre-Beta

Issue Description Class: IfcCostScheduleGroup - Element [IfcCostScheduleElement] -- this does not follow our agreed 'rule of thumb' that all optional lists of 1:N should be changed to mandatory lists of 0:N.

Proposed Solution Make mandatory and change cardinality to L[0:N].

Resolution Agreed..

Action # 1 **Assignee** Wix **Status** Complete **Resolved in Version** R1.5 - Pre-Fin

JW to make the changes. Confirmed in Pre-Final (RS).

Issue Number ***I - 146*** **Issue Date** 8/8/97

Author See **Owner** Wix **Status** Rejected

Schema IfcDocumentExt **Version** R1.5 - Pre-Beta

Issue Description Class: IfcCostSchedule - ApprovedBy [IfcPerson] -- Approvals may come from a department or group. Additionally, approvals may come from a list of people or groups.

Proposed Solution Change this attribute to a mandatory L[0:N] [IfcActor].

Resolution Reject. In practice, a person approves a cost schedule -- someone has to sign it.

Issue Number ***I - 147*** **Issue Date** 8/8/97

Author See **Owner** Wix **Status** Rejected

Schema IfcDocumentExt **Version** R1.5 - Pre-Beta

Issue Description Class: IfcCostSchedule - PreparedBy, ApprovedBy, SubmittedBy -- These concepts apply for various types of analysis and documents 'workflow'. Therefore, they should be generalized and referenced.

Proposed Solution create a generic "WorkFlow" schema including these concepts and others appropriate to workflow -- then reference it here.

Resolution Reject. This is a simplified method for cost schedules only in R1.5.

Issue Number	***I - 148***			***Issue Date***	8/8/97
Author	See	***Owner***	Wix	***Status***	Resolved
Schema	IfcDocumentExt	***Version***	R1.5 - Pre-Beta		

Issue Description Class: IfcCostScheduleElement - ExtensionCost [IfcCostResource.IfcCost] -- is this really needed -- it is simple math, ElementCost x Quantity = ExtensionCost.

Proposed Solution Consider: eliminating this attribute for efficiency.

Resolution Parial agreement. This attribute can be derived (DER).

Action # 1 ***Assignee*** Wix ***Status*** Complete ***Resolved in Version*** R1.5 - Pre-Fin

JW to make the changes. Confirmed in Pre-Final (RS).

Issue Number	***I - 149***			***Issue Date***	8/8/97
Author	See	***Owner***	Wix	***Status***	Resolved
Schema	IfcDocumentExt	***Version***	R1.5 - Pre-Beta		

Issue Description Class: IfcCostScheduleElement - Schedules [IfcKernel.IfcProduct] -- naming is ambiguous.

Proposed Solution Rename to "ProductsCosted".

Resolution Will make "SchedulesProducts".

Action # 1 ***Assignee*** Wix ***Status*** Complete ***Resolved in Version*** R1.5 - Pre-Fin

JW to make the changes. Confirmed in Pre-Final (RS).

Issue Number	***I - 150***			***Issue Date***	8/8/97
Author	See	***Owner***	Wix	***Status***	Rejected
Schema	IfcDocumentExt	***Version***	R1.5 - Pre-Beta		

Issue Description Class: IfcTable - Heading [IfcTableHeading] -- naming.

Proposed Solution Rename to "TableHeadings".

Resolution Rejected..

Issue Number	***I - 151***			***Issue Date***	8/8/97
Author	See	***Owner***	Wix	***Status***	Resolved
Schema	IfcDocumentExt	***Version***	R1.5 - Pre-Beta		

Issue Description Class: IfcTableHeading - HeadingDescriptions [STRING] -- name seems redundant (Heading and Description). Also, the cardinality should not be linked to a value in another object unless this one is to be contained only (violates encapsulation).

Proposed Solution 1) rename to "TableHeadings", 2) change cardinality of the Array to [1:N].

Eliminate these classes and roll them in as attributes of the IfcTable.

Resolution Agreed.

Action # 1 ***Assignee*** Wix ***Status*** Complete ***Resolved in Version*** R1.5 - Pre-Fin

JW/RS/TL will discuss and work out how to eliminate the classes for row and headings. JW has included his compromise solution in release for 8-Sep. Not confirmed in Pre-Final (RS email to JW, 15-Sep).

Action # 2 ***Assignee*** See ***Status*** Complete ***Resolved in Version*** R1.5 - Pre-Fin

JW/RS/TL will discuss and work out how to eliminate the classes for row and headings. JW has included his compromise solution in release for 8-Sep. Not confirmed in Pre-Final (RS email to JW, 15-Sep

Action # 3 **Assignee** Liebich **Status** Complete **Resolved in Version** R1.5 - Pre-Fin

JW/RS/TL will discuss and work out how to eliminate the classes for row and headings. JW has included his compromise solution in release for 8-Sep. Not confirmed in Pre-Final (RS email to JW, 15-Sep

---

Issue Number I - 152 **Issue Date** 8/8/97

Author See **Owner** Wix **Status** Resolved

Schema IfcDocumentExt **Version** R1.5 - Pre-Beta

Issue Description Class: IfcTableHeading - This is a single attribute class -- it could/should be eliminated.

Proposed Solution Convert the "TableHeadings" attribute from IfcTable to be and Array [1:Number of Columns] [STRING].

Resolution Agreed.

Action # 1 **Assignee** Wix **Status** Complete **Resolved in Version** R1.5 - Pre-Fin

JW/RS/TL will discuss and work out how to eliminate the classes for row and headings. Not confirmed in pre-final (RS) - done differently and has some new problems - see new issue on this somewhere after I-215.

Action # 2 **Assignee** See **Status** Complete **Resolved in Version** R1.5 - Pre-Fin

JW/RS/TL will discuss and work out how to eliminate the classes for row and headings. Not confirmed in pre-final (RS) - done differently and has some new problems - see new issue on this somewhere after I-215.

Action # 3 **Assignee** Liebich **Status** Complete **Resolved in Version** R1.5 - Pre-Fin

JW/RS/TL will discuss and work out how to eliminate the classes for row and headings. Not confirmed in pre-final (RS) - done differently and has some new problems - see new issue on this somewhere after I-215.

Issue Number I - 153 **Issue Date** 8/8/97

Author See **Owner** Wix **Status** Resolved

Schema IfcDocumentExt **Version** R1.5 - Pre-Beta

Issue Description Class: IfcTableRow - ValueComponent [IfcMeasureResource.IfcMeasureValue] -- name is a bit too generic and cardinality should not be linked to an attribute in another object (NumberOfColumns) unless this one is to be contained only (violates encapsulation).

Proposed Solution 1) rename "ValueComponent" to "RowValues"

2) change cardinality of this Array to [1:N] OR move "NoOfCellsInRow" attribute into this class (IfcTableRow) from IfcTable.

Resolution Agreed - will make "RowValues" a LIST [1:?].

Action # 1 **Assignee** Wix **Status** Complete **Resolved in Version** R1.5 - Pre-Fin

JW/RS/TL will discuss and work out how to eliminate the classes for row and headings.

Made "RowValues" a LIST [1:?].

Action # 2 **Assignee** See **Status** Complete **Resolved in Version** R1.5 - Pre-Fin

JW/RS/TL will discuss and work out how to eliminate the classes for row and headings. Not confirmed in pre-final (RS email to JW, 15-Sep) - simply not done.

Action # 3 **Assignee** Liebich **Status** Complete **Resolved in Version** R1.5 - Pre-Fin

JW/RS/TL will discuss and work out how to eliminate the classes for row and headings. Not confirmed in pre-final (RS email to JW, 15-Sep) - simply not done.

Issue Number I - 154 **Issue Date** 8/8/97

Author See **Owner** Liebich **Status** Rejected

Schema IfcSharedBldgElements **Version** R1.5 - Pre-Beta

Issue Description General issues for this schema - Missing class: IfcCeiling (from R1.0) is no longer included in the R1.5 model.

Proposed Solution Add it back in -- either as a subtype of IfcCovering or as a subtype of IfcBuildingElement.

Resolution Rejected. This was removed at the request of the implementers in the January Munich meeting.

Issue Number	***I - 155***			***Issue Date***	8/8/97
Author	See	***Owner***	Liebich	***Status***	Resolved
Schema	IfcSharedBldgElements	***Version***	R1.5 - Pre-Beta		

Issue Description Class: IfcWall, IfcFloor, IfcRoofSlab - Redundant attributes -- all three of these classes have exactly the same attributes with the exception of the data type for GenericType. This provides an argument for shared implementation through a supertype. This supertype existed in R1.0 in the LayeredElement. Now we we will encouraging redundant implementations. See also the issued for IfcCovering.

Proposed Solution Re-introduce a supertype (possibly called "IfcLayeredBldgElement" which allows sharing of these attributes (and implementation). Subtype these classes from it.

Complication: this would re-introduce another layer in the model.

Resolution In order to avoid the extra layer -- will introduce an new class called "IfcMaterialLayerSetParameters"

Action # 1 ***Assignee*** Liebich ***Status*** Complete ***Resolved in Version*** R1.5 - Pre-Fin

TL will add new class in IfcPropertiesResource and then reference from inside of IfcWall, IfcFloor, IfcRoofslab and IfcCoveringElement. Confirmed in pre-final (RS) - although with some problems (see new issues on IfcmaterialLayerSetUsage - after #215).

Issue Number	***I - 156***			***Issue Date***	8/8/97
Author	See	***Owner***	Liebich	***Status***	Resolved
Schema	IfcSharedBldgElements	***Version***	R1.5 - Pre-Beta		

Issue Description Class: IfcWall, IfcFloor, IfcRoofSlab - MaterialLayerSetSense [BOOLEAN] -- naming is ambiguous -- when it could be so clear.

Proposed Solution Rename to "MaterialLayerSetLtoR" (LtoR = Left to Right).

Resolution Agreed.

Action # 1 ***Assignee*** Liebich ***Status*** Complete ***Resolved in Version*** R1.5 - Pre-Fin

TL to make changes. Done as "MlsSetLtoR").

Issue Number	***I - 157***			***Issue Date***	8/8/97
Author	See	***Owner***	Liebich	***Status***	Resolved
Schema	IfcSharedBldgElements	***Version***	R1.5 - Pre-Beta		

Issue Description Class: IfcWall, IfcFloor, IfcRoofSlab - calcTotalWidth [IfcPositiveLengthMeasure] -- naming is ambiguous -- what is really meant here is the "thickness" of the wall. "Width" is normally used to refer to the measure left to right when facing a wall segment.

Proposed Solution Rename to "calcTotalThickness".

Resolution Agreed.

Action # 1 ***Assignee*** Liebich ***Status*** Complete ***Resolved in Version*** R1.5 - Pre-Fin

TL will change in the new class defined in the resolution to I-155. Done in IfcMaterialLayerSetUsage (referenced by these classes).

Issue Number	***I - 158***			***Issue Date***	8/8/97
Author	See	***Owner***	Liebich	***Status***	Resolved
Schema	IfcSharedBldgElements	***Version***	R1.5 - Pre-Beta		

Issue Description Class: IfcWall, IfcFloor, IfcRoofSlab - MaterialLayerSetOffset [IfcLengthMeasure] -- naming is ambiguous -- what is really meant here is the MaterialLayerSet (MLS) offset from the Baseline (which is analoguous to the extrusion path defined in the ShapeRep.

Proposed Solution Rename to "MlsOffsetFromBaseline".

Resolution Will change to "MMlsOffsetFromBaseline".

Action # 1 **Assignee** Liebich **Status** Complete **Resolved in Version** R1.5 - Pre-Fin

TL will make the change. Confirmed (RS).

Issue Number I - 159 **Issue Date** 8/8/97

Author See **Owner** Liebich **Status** Resolved

Schema IfcSharedBldgElements **Version** R1.5 - Pre-Beta

Issue Description Class: IfcBuiltIn - Material [IfcMaterial] -- this attribute does not make sense for a Built-In because these are normally assemblies.

Proposed Solution Remove the attribute.

Resolution Agreed.

Action # 1 **Assignee** Liebich **Status** Complete **Resolved in Version** R1.5 - Pre-Fin

TL will make the change. Confirmed in pre-final (RS).

Issue Number I - 160 **Issue Date** 8/8/97

Author See **Owner** Liebich **Status** Resolved

Schema IfcSharedBldgElements **Version** R1.5 - Pre-Beta

Issue Description Class: IfcUserDefBuildingElement - this class is redundant with IfcProxy class currently being discussed.

Proposed Solution Remove it, but be sure to include IfcProxy as has been discussed.

Resolution Agreed.

Action # 1 **Assignee** Liebich **Status** Complete **Resolved in Version** R1.5 - Pre-Fin

TL will make the change. Confirmed in pre-final (RS).

Issue Number I - 161 **Issue Date** 8/8/97

Author See **Owner** Liebich **Status** Resolved

Schema IfcSharedBldgElements **Version** R1.5 - Pre-Beta

Issue Description Class: IfcCovering - This class has the exact same attributes as for IfcWall, IfcFloor, IfcRoofslab. This further supports the notion of a superclass which allow sharing of these attributes and their implementation.

Proposed Solution Re-introduce a supertype (possibly called "IfcLayeredBldgElement" which allows sharing of these attributes (and implementation). Subtype this class from it.

Resolution The solution described in I-155 will be used here as well.

Action # 1 **Assignee** Liebich **Status** Complete **Resolved in Version** R1.5 - Pre-Fin

TL will make the changes. Confirmed in pre-final (RS). See solution to I-155

Issue Number I - 162 **Issue Date** 8/8/97

Author See **Owner** Liebich **Status** Resolved

Schema IfcSharedBldgElements **Version** R1.5 - Pre-Beta

Issue Description Class: IfcCovering - MaterialLayerSetSense [BOOLEAN] -- naming is ambiguous -- when it could be so clear.

Proposed Solution Rename to "MaterialLayerSetLtoR" (LtoR = Left to Right).

Resolution Agreed. See also the discussion in I-155.

Action # 1 ***Assignee*** Liebich ***Status*** Complete ***Resolved in Version*** R1.5 - Pre-Fin

TL will make the changes. Done in IfcMaterialLayerSetUsage (referenced by this class).

Issue Number ***I - 163*** ***Issue Date*** 8/8/97

Author See ***Owner*** Liebich ***Status*** Resolved

Schema IfcSharedBldgElements ***Version*** R1.5 - Pre-Beta

Issue Description Class: IfcCovering - MaterialLayerSetOffset [IfcLengthMeasure] -- naming is ambiguous -- what is really meant here is the MaterialLayerSet (MLS) offset from the Baseline (which is analoguous to the extrusion path defined in the ShapeRep.

Proposed Solution Rename to "MlsOffsetFromBaseline".

Resolution Agreed. See also the discussion in I-155.

Action # 1 ***Assignee*** Liebich ***Status*** Complete ***Resolved in Version*** R1.5 - Pre-Fin

TL will make the changes. Done in IfcMaterialLayerSetUsage (referenced by this class).

Issue Number ***I - 164*** ***Issue Date*** 8/8/97

Author See ***Owner*** Liebich ***Status*** Deferred to R2.0

Schema IfcSharedBldgElements ***Version*** R1.5 - Pre-Beta

Issue Description Class: IfcRelCoversBldgElements - RelatingObject, RelatedObjects -- currently, this shows an IfcBuildingElement as the 'driver' of this '1toN' relationship (it 'has 1toN Coverings"), but this could be the other way around -- that is, there could be an IfcCovering which "covers 1 to N Building Elements". Therefore, I would assert that this relationship should not be to IfcBuildingElements, but to ReferenceFaces on those BuildingElements. The Covering will be 'aligned' with these referenece faces (which may be subsets of actual faces of the Buidling Element geometry). Since IfcCovering now has its own geometry (since it is an IfcProduct), this will be possible.

Proposed Solution 1) reverse the direction of this objectified relationship, 2) change the data type for the RelatedObjects L[1:N] --> IfcModelingAids.IfcReferenceFace ,3) add a set of IfcControls which provide for alignment Points, Curves and Faces to the 'Reference' set in IfcModelingAids -- such alignment classes would allow for any fixed offset from the reference entity.

Resolution Part 1 is rejected -- this relationship direction is consistent with the Space to SpaceBoundary relationship. NOTE: this _can_ be done either way, but we need to do it consistently in the 3 or 4 places where the relationships are essentially 'many to many'. In this case, if the covering covers multiple BuildingElements, each will have a relationship to the covering. Each building element may, by the current direction, relate to multiple coverings.

Parts 2 and 3 are deferred to R2.0.

Action # 1 ***Assignee*** See ***Status*** Complete ***Resolved in Version*** R2.0 - Alpha

Parts 2 and 3 are deferred to R2.0. RS to add to R2.0 STF projects list.

Issue Number ***I - 165*** ***Issue Date*** 8/8/97

Author See ***Owner*** Liebich ***Status*** Resolved

Schema IfcSharedBldgElements ***Version*** R1.5 - Pre-Beta

Issue Description Class: IfcDoor, IfcWindow - With the generalization of the shape representation in R1.5, mapping of semantic meaning to 'components' of the ShapeRep geometry has been lost. Specifically, we no longer have a mapping from attributes in the Semantic Model object for the profiles: TrimA, TrimB, Frame, PanelFrames -- or the overall measures: Thickness, OverallWidth, OverallHeight.

Proposed Solution Provide attributes that are accessible to applications (e.g. simulation apps which need to derive the area of glass versus the area of frame for 'U' value calculations) which drive the actual geometry (through Attribute Driven Geometry ShapeRep).

Resolution 1) Create PropertySets including properties for the Semantic Model objects -- driven by type -- attached to the semantic model object

2) Create an Enum per generic type -- which includes the "Identifiers" for a set of standard Att-Driven ShapeRep components (for this object type) --> these "Identifiers" will be used by the creating app and conformance testing should check these.

3) Add to documentation -- limitation in R1.5 is that the parameters in AttDriven geometry are not yet 'driven' by the properties on the semantic model

4) We will look into including in the documentation -- for each class where the geometry could use attribute driven geometry -- description of the "standard" method for interpreting the semantic model attributes to create the Implicit Geom.

Action # 1 ***Assignee*** See ***Status*** Complete ***Resolved in Version*** R1.5 - Pre-Fin

RS to take on parts 1 & 2. Not confirmed by (RS email, 15-Sep) - this must still be done.

1 complete
2. Must be checked - see enums in I-317.

Action # 2 ***Assignee*** Liebich ***Status*** Complete ***Resolved in Version*** R1.5 - Pre-Fin

TL will take on 3 and 4. Not confirmed by (RS email, 15-Sep) - this must still be done.

Partially complete in late November (see TL email 4-Dec-97) - see also I-317.

Issue Number ***I - 166*** ***Issue Date*** 8/8/97

Author See ***Owner*** Liebich ***Status*** Resolved

Schema IfcSharedBldgElements ***Version*** R1.5 - Pre-Beta

Issue Description Class: IfcDoor, IfcWindow - Attributes should be shared -- most of the attributes driving geometry (described in the last issue) are common to doors and windows -- their implementation should be shared. This can be done through a supertype. In R1.0, this was done through the supertype "IfcFillingElement".

Proposed Solution Create a supertype which defines all of the shared attributes (as described in the previous issue) and subtype Door and Window from it.

Resolution This will be done through PropertySets as described in I-165. Commonly referenced Psets are defined for Frames, Glazing, Hardware, and OpeningFillers (e.g. screening and louvers).

Action # 1 ***Assignee*** See ***Status*** Complete ***Resolved in Version*** R1.5 - Final

RS to handle this with the other PropertySets for Arch.

Issue Number ***I - 167*** ***Issue Date*** 8/8/97

Author See ***Owner*** Forester ***Status*** Resolved

Schema IfcSharedBldgServiceElem ***Version*** R1.5 - Pre-Beta

Issue Description Class: IfcBuildingServiceElement - There is no real need for this class to be defined in the Kernel. It would be more appropriate to move it to this schema --subtyping from a reference to IfcBuildingElement.

Proposed Solution Move this class to this schema and subtype here from a reference to IfcBuildingElement.

Resolution Agreed.

Action # 1 ***Assignee*** Liebich ***Status*** Eliminated ***Resolved in Version*** R1.5 - Pre-Fin

TL/JF to make changes. This supertype was eliminated.

Action # 2 ***Assignee*** Wix ***Status*** Eliminated ***Resolved in Version*** R1.5 - Pre-Fin

TL/JF to make changes. This supertype was eliminated.

Issue Number ***I - 168*** ***Issue Date*** 8/8/97

Author See ***Owner*** Forester ***Status*** Resolved

Schema IfcSharedBldgServiceElem ***Version*** R1.5 - Pre-Beta

Issue Description Class: IfcBuildingServiceElement - Missing Attributes from R1.0 -- This class effectively replaces the IfcManufacturedElement in R1.0. The attributes that were inherited by ElectricalAppliance, Fixture and Equipment are now missing.

Proposed Solution Add the following attributes (from R1.0) to this class: I_BldgServiceElement --> Manufacturer

[IfcActor], ModelLabel [STRING], WarrantyDuration [IfcTimeDuration], OperatingWeight [IfcMassMeasure]; I_Acquisition --> AcquisitionDate [IfcTimeStamp], Supplier [IfcActor], ShippingWeight [IfcMassMeasure].

Resolution This can be handled through an extension PropertySet which is added to Equipment, Fixture, ElectricalAppliance. --> Pset called "Pset_ManufactureInformation" to be included in the IfcProductExtension Schema.

Action #	1	***Assignee***	Forester	***Status***	Complete	***Resolved in Version***	R1.5 - Pre-Fin

JF to add new attributes. Not yet confirmed in pre-final (RS).

Action #	2	***Assignee***	See	***Status***	Complete	***Resolved in Version***	R1.5 - Final

RS to insure that this is also referenced by manufactured elements in the SharedBldgElement, Architecture and FM schemata.

Issue Number I - 169 ***Issue Date*** 8/8/97

Author	See	***Owner***	See	***Status***	Rejected
Schema	IfcArchitecture	***Version***	R1.5 - Pre-Beta		

Issue Description Select type: IfcProgrammeGroupOrSpace - Name is misleading because it can be taken to indicate that one of the choices is IfcSpace.

Proposed Solution Change name to "IfcProgrammeGroupOrSpaceProgramme".

Resolution Rejected because this SelectType had to be removed -- since select types cannot be used in this way in EXPRESS

Issue Number I - 170 ***Issue Date*** 8/8/97

Author	See	***Owner***	See	***Status***	Resolved
Schema	IfcArchitecture	***Version***	R1.5 - Pre-Beta		

Issue Description Use from Schema: IfcActor, IfcSpace - in each of these cases, the "USE" from should be changed to a "Reference" from. Additionally, the schema for IfcActor is IfcPropertyResource, not IfcActorRes.

Proposed Solution Change to "Reference" from and correct error in schema name for IfcActor.

Resolution Agreed.

Action #	1	***Assignee***	See	***Status***	Complete	***Resolved in Version***	R1.5 - Pre-Fin

RS to make the change. Confirmed in pre-final (RS).

Issue Number I - 171 ***Issue Date*** 8/8/97

Author	See	***Owner***	See	***Status***	Resolved
Schema	IfcArchitecture	***Version***	R1.5 - Pre-Beta		

Issue Description Class: IfcRelAdjacencyReq - RelatingObject, RelatingObject[type] -- The attribute "RelatingObject" for the supertype 'Relationship1to1" is redeclared twice. This cannot be right.

Proposed Solution The one with the INV relationship called "HasAdjacencyReqFrom S[0:N] should be a redeclaration of the "RelatedObject".

Resolution Agreed.

Action #	1	***Assignee***	See	***Status***	Complete	***Resolved in Version***	R1.5 - Pre-Fin

RS to make the change. Confirmed in pre-final (RS).

Issue Number I - 172 ***Issue Date*** 8/8/97

Author	See	***Owner***	Forester	***Status***	Resolved
Schema	IfcHVAC	***Version***	R1.5 - Pre-Beta		

Issue Description Class: IfcFluidMover - DataTypes incorrect -- As stated in the R1.0 specifications, the data type for many of the attributes on this class should be updated to use the new measure schema.

Proposed Solution FlowRate [IfcFlowRateMeasure], WorkingPressure [IfcPressureMeasure], OperatingEfficiency/MinimumEfficiency [IfcPercentageMeasure], OperatingPower/MaximumPower [IfcEnergyMeasure], Speed [IfcVelocityMeasure(?)].

Resolution Agreed in principle, but these measure types are not included as MeasureValues, therefore, all but Speed will be of type IfcMeasureWithUnit.

Action #	1	***Assignee***	Forester	***Status***	Complete	***Resolved in Version***	R1.5 - Pre-Fin

JF will make changes. Confirmed in pre-final (RS) - except that even Speed was set to date type IfcMeasureWithUnit.

Issue Number ***I - 173*** ***Issue Date*** 8/8/97

Author	See	***Owner***	See	***Status***	Resolved
Schema	IfcFacilitiesMgmt	***Version***	R1.5 - Pre-Beta		

Issue Description Superclass: IfcProductExtension.IfcElement - This should not be a "Use" from schema (if it is to be consistent with our convention).

Proposed Solution Change it to a "Reference" from schema.

Resolution Agreed.

Action #	1	***Assignee***	See	***Status***	Complete	***Resolved in Version***	R1.5 - Pre-Fin

RS will make change. Confirmed in pre-final (RS).

Issue Number ***I - 174*** ***Issue Date*** 8/8/97

Author	See	***Owner***	See	***Status***	Resolved
Schema	IfcFacilitiesMgmt	***Version***	R1.5 - Pre-Beta		

Issue Description Missing Superclass: IfcFacilitiesElement - In R1.0, IfcFuriture derived from IfcManufacturedElement. As this superclass has been eliminated in R1.5, the attributes that were inherited from it must be replaced.

Proposed Solution Add the following attributes (from R1.0) to this class: I_FacilitiesElement --> Manufacturer [IfcActor], ModelLabel [STRING], WarrantyDuration [IfcTimeDuration], OperatingWeight [IfcMassMeasure]; I_Acquisition --> AcquisitionDate [IfcTimeStamp], Supplier [IfcActor], ShippingWeight [IfcMassMeasure].

Resolution Will use the "Pset_ManufactureInformation" propertyset described in I-168 and attach as a Domain View - Type driven OccurrencePropertySet.

All property sets requiring this information must be modified to utilize the common Pset_ManufactureInformation property set

Action #	1	***Assignee***	See	***Status***	Complete	***Resolved in Version***	R1.5 - Final

RS to modify, Product, SharedBldgElement and Architecture Property Sets

Action #	2	***Assignee***	Forester	***Status***	Complete	***Resolved in Version***	R1.5 - Pre-Fin

JF to modify HVAC Property Sets

Action #	3	***Assignee***	Yu	***Status***	Complete	***Resolved in Version***	R1.5 - Final

KY to modify FM Property Sets

Issue Number ***I - 175*** ***Issue Date*** 8/8/97

Author	See	***Owner***	See	***Status***	Rejected
Schema	IfcFacilitiesMgmt	***Version***	R1.5 - Pre-Beta		

Issue Description Class: IfcFurniture - AssignedTo [IfcActor] -- this attribute should be mandatory.

Proposed Solution Make it mandatory.

Resolution Rejected. You may not know to whom it belongs.

Issue Number	I - 176			**Issue Date**	8/8/97
Author	See	**Owner**	See	**Status**	Resolved
Schema	IfcFacilitiesMgmt	**Version**	R1.5 - Pre-Beta		

Issue Description Class: IfcFurniture - Condition [STRING, MainColor [STRING], PhysicalVolume [IfcVolumeMeasure] -- these attributes should be optional as they may not be know.

Proposed Solution Make them optional.

Resolution Changed -- MainColor, PhysicalVolume, Condition will be made optional.

Action # 1 **Assignee** See **Status** Complete **Resolved in Version** R1.5 - Pre-Fin

RS to make changes. Confirmed in pre-final (RS).

Issue Number	I - 177			**Issue Date**	7/28/97
Author	Forester	**Owner**	Liebich	**Status**	Resolved
Schema	IfcMeasureResource	**Version**	R1.5 - Pre-Beta		

Issue Description Missing specific unit types for attributes in the IfcHvac Schema.

Proposed Solution Need to add the following specific unit measure support in IfcMeasureResource:

IfcVolumetricFlowrateMeasure - REAL (m3/s)
IfcMassFlowrateMeasure - REAL (kg/s)
IfcPercentMeasure - REAL (Unitless: range 0 - 1.0000) <-- this is ratio
IfcPressureMeasure - REAL (Pa)
IfcEnergyMeasure - REAL (J)
IfcPowerMeasure - REAL (W)
IfcAngularVelocityMeasure - REAL (rad/s)
IfcLinearVelocityMeasure - REAL (m/s)
IfcRotationalFrequencyMeasure - REAL (rev/s)
IfcHeatfluxDensityMeasure - REAL (W/m2)
IfcMassDensityMeasure - REAL (kg/m3)
IfcThermalAdmittanceMeasure - REAL
IfcThermalResistanceMeasure - REAL (m2 K / W)
IfcThermalTransmittanceMeasure - REAL (W/m2 K)
IfcVoltageMeasure - REAL (V)
IfcDynamicViscosityMeasure - REAL (Pa s)
IfcKinematicViscosityMeasure - REAL (m2/s)

Resolution Solution:
1) add enumeration to the IfcDerivedUnit (IfcDerivedUnitEnum) which includes these
2) do not add these to the IfcMeasureValue select type, but use the IfcPropertyWithUnit in Properties and PropertySets instead.

Action # 1 **Assignee** Liebich **Status** Complete **Resolved in Version** R1.5 - Pre-Fin

TL will make changes to the Measure schema. Confirmed (RS).

Issue Number	I - 178			**Issue Date**	8/15/97
Author	Forester	**Owner**	Liebich	**Status**	Resolved
Schema	IfcGeometryResource	**Version**	R1.5 - Pre-Beta		

Issue Description IfcBoundingBox has attributes of Z,Y,Z in EXPRESS-G

Proposed Solution Should be X,Y,Z

Resolution Already resolved by TL

Issue Number	I - 179			**Issue Date**	8/15/97
Author	Forester	**Owner**	Liebich	**Status**	Rejected
Schema	IfcPropertyTypeResource	**Version**	R1.5 - Pre-Beta		

Issue Description Does not appear to be any relationship between IfcPropertySet and IfcSimpleProperty

Proposed Solution There should be a relationship here

Resolution Rejected. This is already in the model -- it is one level up - in the relationship to the supertype IfcPropertyDef.

Issue Number ***I - 180*** ***Issue Date*** 8/7/97

Author Shulga ***Owner*** Liebich ***Status*** Resolved

Schema IfcGeometryResource ***Version*** R1.5 - Pre-Beta

Issue Description Geometry is geometry is geometry

Proposed Solution All geometry entities should be derived from IfcGeometryRepresentationItem -- including the AttDrivenGeom Profile, Path, ExtrusionSolid entities

Resolution Agreed.

Action # 1 ***Assignee*** Liebich ***Status*** Complete ***Resolved in Version*** R1.5 - Pre-Fin

TL to make the change. Confirmed in pre-final (RS).

Issue Number ***I - 181*** ***Issue Date*** 8/7/97

Author Shulga ***Owner*** Liebich ***Status*** Deferred to R2.0

Schema IfcGeometryResource ***Version*** R1.5 - Pre-Beta

Issue Description Numeric precision of B-Reps is undefined in IFC

Proposed Solution Someone should study this and define it. Nikolay has volunteered to help.

Resolution Short term solution is inclusion of attribute "Precision" on IfcRepresentationContext. Long term solutions deferred for inclusion in R2.0.

Action # 1 ***Assignee*** Liebich ***Status*** Incomplete ***Resolved in Version*** R2.0 - Alpha

TL to work with Nikolay Shulga to investigate and make recommendations.

Action # 2 ***Assignee*** See ***Status*** Complete ***Resolved in Version*** R2.0 - Alpha

RS to add to R2.0 STF projects list.

Issue Number ***I - 182*** ***Issue Date*** 8/7/97

Author Shulga ***Owner*** Liebich ***Status*** Resolved

Schema IfcGeometryResource ***Version*** R1.5 - Pre-Beta

Issue Description There is no geometry supertype for attribute driven solids

Proposed Solution Should be subtyped from IfcSolidModel

Resolution Agreed.

Action # 1 ***Assignee*** Liebich ***Status*** Complete ***Resolved in Version*** R1.5 - Final

Make the change as proposed.

Issue Number ***I - 183*** ***Issue Date*** 8/7/97

Author Shulga ***Owner*** Liebich ***Status*** Resolved

Schema IfcGeometryResource ***Version*** R1.5 - Pre-Beta

Issue Description IfcAttributeDriven profile definition uses a different mechanism for placement

Proposed Solution Should use IfcAxisPlacement

Resolution Agreed. PosX, PosY and Alpha will be replaced by a single attribute called Placement (of type IfcAxisPlacement2D.

Action # 1 ***Assignee*** Liebich ***Status*** Complete ***Resolved in Version*** R1.5 - Pre-Fin

TL to make changes with help from Nikolay on the changing the functions for creating the xxxResolution geometry in the subtypes. Confirmed in pre-final (RS) - except that the attribute is called Position (to be consistent with the rest of geometry.

Issue Number	**I - 184**			**Issue Date**	8/15/97
Author	Forester	**Owner**	Liebich	**Status**	Resolved
Schema	IfcKernel	**Version**	R1.5 - Pre-Beta		

Issue Description There is a problem for implementers w/o IfcProxy

Proposed Solution We need to include the proposed IfcProxy

Resolution Agreed -- will use the generalized Proxy proposed by TL in email on 8/6/97

Single IfcProxy
-> subtype proxy from IfcObject, use "ExtendedProperties" to attach the appropriate properties. Since all predefined properties (like IfcCost, IfcActor, ...) are now subtyped from IfcPropertyDef they can be handled by those dynamic lists.

Pros: very flexible

Cons: needs solution for shape (but this can be tight to the other issueto consider shape as being just another property under IfcPropertyDef)

TL prefers the last alternative:

```
ENTITY IfcProxy
   SUBTYPE FROM (IfcObject);
       ProxyType      : IfcProxyTypeEnum;
       LocalPlacement : OPTIONAL IfcLocalPlacement;
       ResultsIn : OPTIONAL  IfcSequence;
(*  Solution for ProductShape *)
WHERE
    WR1 : NOT (EXISTS (SELF\IfcObject.TypeDefinedProperty));
    WR2 : HIINDEX (OccurrenceProperties) = 0;
END_ENTITY;
```

Action #	1	**Assignee**	Liebich	**Status**	Complete	**Resolved in Version**	R1.5 - Pre-Fin

TL will make changes in the Kernel. Confirmed inpre-final (RS).

Issue Number	**I - 185**			**Issue Date**	8/21/97
Author	Liebich	**Owner**	Liebich	**Status**	Resolved
Schema	IfcTypeDefResource	**Version**	R1.5 - Pre-Beta		

Issue Description Currently, both "Generic" and "Specific" PropertySets are optional

Proposed Solution We need a constraint that either a "Generic" or "Specific" type will be defined

Resolution Agreed -- this will be done with a WHERE rule on IfcPropertyTypeDef.

Action #	1	**Assignee**	Liebich	**Status**	Complete	**Resolved in Version**	R1.5 - Final

TL to make the change.

Issue Number	**I - 186**			**Issue Date**	8/21/97
Author	Liebich	**Owner**	Liebich	**Status**	Resolved
Schema	IfcKernel	**Version**	R1.5 - Pre-Beta		

Issue Description Currently, we can add simple properties directly through OccurrenceProperties and ExtendedProperties

Proposed Solution Change the data type for both from IfcPropertyDef to IfcPropertySet

Resolution Agreed. This will be changed to allow attachment of PropertySets only.

Action #	1	**Assignee**	Liebich	**Status**	Complete	**Resolved in Version**	R1.5 - Pre-Fin

TL to make the change. Confirmed inpre-final (RS).

Issue Number I - 187 Issue Date 8/21/97

Author Liebich Owner Liebich Status Resolved

Schema IfcTypeDefResource Version R1.5 - Pre-Beta

Issue Description Currently, IfcPropertySet does not have to have any properties (list of 0:?)

Proposed Solution Change the cardinality of the list to 1:?

Resolution Agreed.

Action # 1 Assignee Liebich Status Complete Resolved in Version R1.5 - Final

TL to make the change. Not confirmed inpre-final (RS email to TL, 15-Sep).

Issue Number I - 188 Issue Date 8/21/97

Author Liebich Owner Liebich Status Deferred to R2.0

Schema IfcTypeDefResource Version R1.5 - Pre-Beta

Issue Description Currently the uniqueness of simple properties is not defined.

Proposed Solution Add a unique label which insures that each simple property is uniquely defined and understood.

Resolution Defer discussion and proposed solution until R2.0.

Action # 1 Assignee See Status Complete Resolved in Version R2.0 - Alpha

RS to add to R2.0 STF projects list.

Issue Number I - 189 Issue Date 8/21/97

Author Liebich Owner Liebich Status Resolved

Schema IfcSharedBldgElements Version R1.5 - Pre-Beta

Issue Description We do not have any information to resolve layered wall connection (e.g. the layer priority problem)

Proposed Solution Do it!

Resolution 1) We will introduce a table of FundamentalMaterials (7 are currently defined in Germany - which seem to be appropriate to all countries).

2) Will add an Priority Index to the MaterialLayerSet. The order in which the layers should be connected to the other wall.

3) We will add an optional Array of a pair of Integers -- called ConnectionOverrides.

Action # 1 Assignee Liebich Status Complete Resolved in Version R1.5 - Pre-Fin

TL and RJ will make the changes. Confirmed (RS).

Action # 2 Assignee Junge Status Complete Resolved in Version R1.5 - Pre-Fin

TL and RJ will make the changes. Confirmed (RS).

Issue Number I - 190 Issue Date 8/21/97

Author Forester Owner Forester Status Resolved

Schema IfcSharedBldgServiceElem Version R1.5 - Pre-Beta

Issue Description There is no reference to Material for the subtypes of BuildingElements

Proposed Solution Add material references

Resolution Done

Issue Number I - 191 Issue Date 8/21/97

Author See Owner Liebich Status Deferred to R2.0

Schema IfcGenericResource Version R1.5 - Pre-Beta

Issue Description Class: IfcOwnerIdentification.OwningActor - This is an add-on issue related to I-001. Resolution to that issue resulted in a simple list of Actors referenced by this attribute (now an integer index into the ProjectTeamRegistry). This issue is to add enhancements to the ProjectTeamRegistry by incorporating a model for standard roles in project processes (e.g. workflow control). This would allow application developers to incorporate workflow messaging (e.g. Architect reaches "Arch. Concept Design" milestone and submits to shared model with messages to "Structural Engr" and "HVAC Engr" project roles that they are next in line to create their correstponding "Concept Design"s. This messaging could then be routed to the appropriate team member -- based on who has been assigned these roles in the Project Team Registry. NOTE: I am not suggesting that we include workflow features in R1.5 or even in R2.0, but that a project team registry would be essential to such things in the future, so let's structure for it now and not have to re-structure later.

Proposed Solution Include a "ProjectRole" for each actor in the project team registry and think about how this could be used for workflow management within the design team. Note: this is different than the document oriented workflow done by products like WorkCenter -- this is workflow in the design process - independent of particular documents.

Resolution This was partially resolved in I-001, workflow and project roles ideas through a more complete, general purpose registry deferred to this issue.

Workflow and project roles related enhancements deferred to R2.0

Action #	1	***Assignee***	See	***Status***	Complete	***Resolved in Version***	R2.0 - Alpha

Workflow and project roles related enhancements deferred to R2.0. RS to add to R2.0 STF projects list.

Issue Number ***I - 192*** ***Issue Date*** 8/22/97

Author	Forester	***Owner***	Liebich	***Status***	Resolved
Schema	IfcProductExt	***Version***	R1.5 - Pre-Beta		

Issue Description The IfcBuildingSection and IfcBuildingSubStorey could be represented by Zones.

Proposed Solution Eliminate these classes and use IfcZone instead.

Resolution Agreed in principal -- but investigation first.

Action #	1	***Assignee***	Wix	***Status***	Complete	***Resolved in Version***	R1.5 - Pre-Fin

JW will do some investigation with Steve Race for his input (based on his experience in developing Oxes and BDS).

Issue Number ***I - 193*** ***Issue Date*** 7/15/97

Author	Haiat	***Owner***	Liebich	***Status***	Resolved
Schema	IfcProductExt	***Version***	R1.5 - Pre-Beta		

Issue Description I would like to have an ordered list of SpaceBoundaries for each Space.

Proposed Solution Reverse the relationship between Space and SpaceBoundary and make it a list.

Resolution Agreed.

Action #	1	***Assignee***	Liebich	***Status***	Complete	***Resolved in Version***	R1.5 - Pre-Fin

TL will make the change. Confirmed in pre-final (RS).

Issue Number ***I - 194*** ***Issue Date*** 8/22/97

Author	Liebich	***Owner***	Liebich	***Status***	Resolved
Schema	IfcGeometryResource	***Version***	R1.5 - Pre-Beta		

Issue Description Currently cannot differentiate the use for multiple alternative shape representations.

Proposed Solution Need to add a "Usage" attribute on the IfcShapeRepresentation so that we can identify what the shape represents -- e.g. this one represents site boundaries, that one represents countours, last one represents ground form.

Resolution Agreed.

Action #	1	***Assignee***	Liebich	***Status***	Complete	***Resolved in Version***	R1.5 - Pre-Fin

TL will make the addition. Confirmed in pre-final (RS).

Issue Number   ***I - 195***   ***Issue Date*** 8/23/97

Author	Forester	***Owner***	Liebich	***Status***	Resolved
Schema	IfcSharedBldgElements	***Version***	R1.5 - Pre-Beta		

Issue Description IfcCoveringElement is missing the GenericType to drive the TypeDefinition.

Proposed Solution Add GenericType in

Resolution Agreed.

Action #	1	***Assignee***	Liebich	***Status***	Complete	***Resolved in Version***	R1.5 - Pre-Fin

TL will make the addition. Confirmed in pre-final (RS).

Issue Number   ***I - 196***   ***Issue Date*** 8/21/97

Author	Wix	***Owner***	Liebich	***Status***	Deferred to R2.0
Schema	IfcGeometryResource	***Version***	R1.5 - Pre-Beta		

Issue Description Limiting IfcMorphingExtrusionSegment to the same profile type at start and end is too limiting. An example would be rectangular to round duct transitions.

Proposed Solution Support different profiles and profiles with different numbers of verticies.

Resolution This will be deferred to R2.0

Action #	1	***Assignee***	See	***Status***	Complete	***Resolved in Version***	R2.0 - Alpha

RS to add to R2.0 STF projects list.

Issue Number   ***I - 197***   ***Issue Date*** 8/23/97

Author	See	***Owner***	See	***Status***	Resolved
Schema	IfcModelingAidExt	***Version***	R1.5 - Pre-Beta		

Issue Description It should not be possible to TypeDef an IfcModelingAids.

Proposed Solution Subtype from IfcRoot instead.

Resolution Agreed.

Action #	1	***Assignee***	Liebich	***Status***	Complete	***Resolved in Version***	R1.5 - Pre-Fin

TL will make the change to Kernel. Confirmed (RS).

Issue Number   ***I - 198***   ***Issue Date*** 8/19/97

Author	Yu	***Owner***	Wix	***Status***	Resolved
Schema	IfcMeasureResource	***Version***	R1.5 - Pre-Beta		

Issue Description IfcTimeDuration - I believe we need two entities to represent time period: one is IfcTimeDuration as defined in the current version. I however would prefer to rename it as IfcTimePeriod since it does represent a specific period of time. The other one is an entity that represents a longevity of time.

Proposed Solution The following is my proposal.

```
ENTITY IfcTimePeriod
   StartTime: IfcDateTimeSelect;
   EndTime: OPTIONAL IfcDateTimeSelect;
END_ENTITY;
ENTITY IfcTimeDuration
     TimeDuration: IfcTimeMeasure;  //could also be IfcTimeUnit, see below
END_ENTITY;
```

```
TYPE IfcTimeUnit = SELECT (Second, Minute, Hour, Day, Week, Month, Quarter, Year);
END_TYPE;
```
(* I understand Thomas's concern about 1week 2 days problem. I think we can deal with this by conversion functions in later release *)

Resolution Have added IfcTimeDurationMeasure and a time measure unit in the IfcUnitTypeEnum. This allows measure of time duration.

Issue Number	***I - 199***			***Issue Date***	8/19/97
Author	Yu	***Owner***	Wix	***Status***	Resolved
Schema	IfcProcessExt	***Version***	R1.5 - Pre-Beta		

Issue Description

IfcWorkSchedule - In the model document, the description of TotalFloat of IfcWorkSchedule has the following statement:
Float time may be either positive, zero or negative. Where it is zero or negative, the task becomes critical.
I think a more accurate description would be:
Free float time may be either positive, zero or negative. Total float time my be either positive or zero. Where the total float is zero, the task becomes critical.

The following definitions are for reference or documentation:
Total Float is: the amount of time that an activity can be delayed without affecting the final duration of the project. (the current description about total float is good too).

Free Float is: the maximum amount of time that an activity can be delayed without having any other effect on the activities around it.

There are other types of activity floats but these 2 are the fundamental ones.

Proposed Solution

```
ENTITY IfcWorkTaskSchedule;      //is renamed as suggested by Richard
    ProjectId        : IfcProjectUniqueId;
    ActualStart      : OPTIONAL IfcDateTimeSelect;
    EarliestFinish   : OPTIONAL IfcDateTimeSelect;
    LatestFinish     : OPTIONAL IfcDateTimeSelect;
    ActualFinish     : OPTIONAL IfcDateTimeSelect;
    EarliestStart    : OPTIONAL IfcDateTimeSelect;
    LatestStart      : OPTIONAL IfcDateTimeSelect;
    StatusTime       : OPTIONAL IfcDateTimeSelect;
    ScheduledStart   : OPTIONAL IfcDateTimeSelect;
    ScheduledFinish  : OPTIONAL IfcDateTimeSelect;
    ScheduleDuration : OPTIONAL IfcTimeDuration;    //use new data type
RemainingTime    : OPTIONAL IfcTimePeriod         //use new data type
    TotalFloat       : OPTIONAL IfcTimeDuration;         //use new data type
    FreeFloat        : OPTIONAL IfcTimeDuration;          //new added attribute
ActualDuration          :  OPTIONAL IfcTimePeriod;  //new added attribute
IsCritical       : OPTIONAL BOOLEAN;
TaskStatus       : OPTIONAL IfcTaskStatusEnum;
END_ENTITY;
```
(* some of the attributes are derived attributes. A DERIVE clause can be added in later release when enough operation functions are provided for time measuring types *)

Resolution JW to work with KY to improve definitions.

Action #	1	***Assignee***	Wix	***Status***	Complete	***Resolved in Version***	R1.5 - Final

JW/KY to work on final changes.

Action #	2	***Assignee***	Yu	***Status***	Complete	***Resolved in Version***	R1.5 - Final

JW/KY to work on final changes.

Issue Number	***I - 200***			***Issue Date***	8/19/97
Author	Yu	***Owner***	Liebich	***Status***	Resolved
Schema	IfcKernel	***Version***	R1.5 - Pre-Beta		

Issue Description

IfcSequence - There are a few of problems around IfcProcess and IfcSequence.
First, the sequence type (SS, SF, etc.) is missing in IfcSequence. Second, IfcSequence has link to multiple IfcProcesses. This doesn't work for the single value of TimeLag and Sequence type.

In a real construction project, it is not common to see one process link a multiple processes with the same lag and link type. Even though one could happen to find such links, it is not a good idea to model these links with one entity, since CM applications always tend to manipulate each process (i.e.task or activity) and each link individually. Therefore, speaking the models, I don't think it is a good idea for IfcSequence to have multiple links to IfcProcess either directly or as Inverse. Third, it also makes sense to me that IfcSequence is a subtype of IfcRelationship1to1 between a predecessor and a successor, and would like to leave this idea open for discussion.

Proposed Solution Proposed solution - ENTITY IfcSequence

```
SUBTYPE OF (IfcKernelRoot);
   SequenceRelTo    : IfcProcess;
   TimeLag           : IfcTimeDuration;     //use new data type
   SequenceType     : IfcSequenceType;  //new data type, see below
 INVERSE
   IsPredecessorFrom : IfcProcess//note: Set[0:?] is eliminated
       FOR ResultsIn;
END_ENTITY;

TYPE IfcSequenceType = SELECT (
FS, (*represents Finish-Start relationship*)
SS, (*represents Start-Start relationship*)
FF, (*represents Finish-Finish relationship*)
SF) (*represents Start-Finish relationship*)
END_TYPE;
```

Resolution JW to work with KY on final resolution

15-Nov-97: IfcSequence will now subtype from IfcRelationship1to1 - note this means that multiple relationship will have to be created for 1toN and NtoN conditions.

Action # 1 ***Assignee*** Wix ***Status*** Complete ***Resolved in Version*** R1.5 - Final

JW/KY to work on final changes.

Action # 2 ***Assignee*** Yu ***Status*** Complete ***Resolved in Version*** R1.5 - Final

JW/KY to work on final changes.

Issue Number ***I - 201*** ***Issue Date*** 8/19/97

Author Yu ***Owner*** Liebich ***Status*** Resolved

Schema IfcKernel ***Version*** R1.5 - Pre-Beta

Issue Description IfcRelUsesConstructionAids - I was 100% sure about this but I thought this entity was to replace IfcResourceUse. If so, I don't think this entity is correctly modeled it only allows one IfcProcess to link to one IfcRelUsesConstructionAids (as Inv. UsesConstructionAids S[0:1]) which cannot deal with each resource usage individually. If however not for this purpose, we need another entity to represent resource use.

Proposed Solution I would propose the following model in addition to the existing ones or to replace IfcRelUsesConstructionAids. (note: an Inverse relationship needs to be added in IfcProcess accordingly).

```
ENTITY IfcResourceUse;  // or IfcConstructionAidUse
   Usedby: IfcProcess;  //use reference
   Resource: IfcConstructionAid;  //use reference
   Quantity: IfcMeasureValue;
   Duration: IfcTimeMeasure;
   Cost: IfcCost;
END_ENTITY;
```

Resolution IfcRelUsesConstructionAids was eliminated.

KY to double check new IfcResourceUse class and work with JW if does not match up.

Action # 1 ***Assignee*** Wix ***Status*** Complete ***Resolved in Version*** R1.5 - Final

JW/KY to work on final changes.

Action # 2 ***Assignee*** Yu ***Status*** Complete ***Resolved in Version*** R1.5 - Final

JW/KY to work on final changes.

Issue Number	***I - 202***			***Issue Date***	8/19/97
Author	Yu	***Owner***	Liebich	***Status***	Resolved
Schema	IfcProductExt	***Version***	R1.5 - Pre-Beta		

Issue Description IfcSpace, IfcPartialSpace - I don't think the IfcPartialSpace is needed here because of two following reasons: 1). it is a subtype of IfcSpace; 2). it doesn't have any more attributes and that of IfcSpace. What we really want to model here is a containment (i.e. has) relationship between IfcSpace and IfcSpace. I think this is the place where we could use IfcRelationship1ToN.

Proposed Solution I think this is the place where we could use IfcRelationship1ToN. I would propose the following for consideration:

```
ENTITY IfcRelHasSpaces
SUBTYPE OF (IfcRelationship1ToN);
SELF\IfcRelationship1ToN.RelatingObject : IfcSpace;
SELF\Relationship1ToN.RelatedObjects : SET [1:?] OF IfcSpace;
END_ENTITY;

ENTITY IfcRelationship1ToN
(*all the existing attributes, plus the following*)
      SUPERTYPE OF (IfcRelHasSpaces);
END_ENTITY;

ENTITY IfcSpace;
(*all the existing attributes, plus the following*)
  INVERSE
HasSpaces : IfcRelHasSpaces FOR SELF/IfcRelationship1ToN.RelatingObject;
IsPartOfSpace : IfcRelHasSpaces FOR SELF\Relationship1ToN.RelatedObjects;
END_ENTITY;
```

Please note that I use 'SET' in IfcRelHasSpaces. I think it is ok to redeclare the attribute at subtype level using different aggregation data type.

Resolution PartialSpace has been eliminated in favor of allowing nesting of Spaces. KY -- check the new schema.

Action #	1	***Assignee***	Yu	***Status***	Complete	***Resolved in Version***	R1.5 - Final

Check the new schema and inform TL if still have issues.

Issue Number	***I - 203***			***Issue Date***	8/19/97
Author	Yu	***Owner***	See	***Status***	Resolved
Schema	IfcArchitecture	***Version***	R1.5 - Pre-Beta		

Issue Description IfcRelAdjacencyReq - The IfcRelAdjacencyReq is currently associated with IfcSpaceProgramme but not IfcSpace. I think the space adjacency relationship should relate directly to 2 spaces that are adjacent each other. I think this requirement fits for both Architectural and FM.

Proposed Solution I would propose the following models:

```
ENTITY IfcRelAdjacencyReq
SUBTYPE OF (IfcRelationship1To1);
SELF\IfcRelationship1To1.RelatingObject: IfcSpace;
SELF\IfcRelationship1To1.RelatedObject: IfcSpace;
  INVERSE
IsForSpaceProgramme : IfcSpaceProgramme FOR HasAdjacencyReqs;
END_ENTITY;

ENTITY IfcRelationship1To1
(*all the existing attributes, plus the following*)
SUPERTYPE OF (IfcRelAdjacencyReq);
END_ENTITY;

ENTITY IfcSpace;
(*all the existing attributes, plus the following*)
 INVERSE
HasAdjacencyReqFrom : SET[0:?] OF IfcRelAdjacencyReq
FOR   SELF\IfcRelationship1To1.RelatingObject;
HasAdjacencyReqsTo : SET[0:?] OF IfcRelAdjacencyReq
```

```
FOR  SELF\IfcRelationship1To1.RelatedObject;
END_ENTITY;

ENTITY IfcSpaceProgramme;
(*all the existing attributes, plus the following*)
HasAdjacencyReqs : SET [0:?] OF IfcRelAdjacencyReq;
END_ENTITY;
```

Resolution RS and TL are not really convinced. It is indirectly related to the space through its program.

Issue Number ***I - 204*** ***Issue Date*** 8/19/97

Author Yu ***Owner*** See ***Status*** Resolved

Schema IfcArchitecture ***Version*** R1.5 - Pre-Beta

Issue Description IfcSpaceProgramme - SpaceName and SpaceUse attributes are not clearly explained. In the documentation, it says: programme name for and space use required of ‘this’ space. Note, that the space programme links to multiple spaces. What does the ‘this’ refer to? IfcSpace should also have a link or an Inverse link to IfcSpaceProgramme.

Proposed Solution Improve the documentation in these areas.

Resolution Improve the documentation for attribute definitions. Cannot do the inverse relationship because Space is in the ProductExt and cannot upward reference the SpaceProgramme.

Action # 1 ***Assignee*** See ***Status*** Complete ***Resolved in Version*** R1.5 - Final

Improve the documentation as proposed.

Issue Number ***I - 205*** ***Issue Date*** 8/19/97

Author Yu ***Owner*** Liebich ***Status*** Resolved

Schema IfcGenericResource ***Version*** R1.5 - Pre-Beta

Issue Description IfcProjectUniqueID and IfcGloblyUniqueID - I think Richard commented this also. I wasn’t so sure about what was the original purpose of having this two Ids. If both are to represent unique Ids generated by a computer program such as COM, they will be globally unique anyway. If so, why bother have two? But, if IfcProjectUniqueID is for a user to set a project level code for an object, like PROJ001-ACT1, it is fine. We need more explanation in the documentation.

Proposed Solution Improve the documentation in the areas cited.

Resolution Add a better explanation of how the two uniqueIDs are combined to form a global unique ID.

Action # 1 ***Assignee*** Liebich ***Status*** Complete ***Resolved in Version*** R1.5 - Final

improve the documentation of IfcUtilityResource as described.

Issue Number ***I - 206*** ***Issue Date*** 8/19/97

Author Yu ***Owner*** See ***Status*** Resolved

Schema IfcModelingAidExt ***Version*** R1.5 - Pre-Beta

Issue Description IfcModelingAid - I didn’t look into this in very detail, but I have the feeling most IfcModelingAid related entities in the IfcModelingAid Schema are related to design. In this sense, the ‘IfcModeling’ seems a little bit confusing for me.

Proposed Solution Can we call it "IfcDesignAid" ?

Resolution Name change not substantially different. Would prefer not to make the change at this late date.

Issue Number ***I - 207*** ***Issue Date*** 8/23/97

Author See ***Owner*** Liebich ***Status*** Resolved

Schema IfcProductExt ***Version*** R1.5 - Pre-Beta

Issue Description [raised by Peter Muigg - issue logged by R.See]

IfcSpace - this class is missing an attribute for the Height of the Space. This is needed in order to calculate the volume. While it may be possible to deduce this from the

Proposed Solution Add an attribute for Height

Resolution Agreed -- called "calcAverageHeight"

Action # 1 **Assignee** Liebich **Status** Complete **Resolved in Version** R1.5 - Final

TL to add it

Issue Number **I - 208** **Issue Date** 8/23/97

Author Liebich **Owner** Liebich **Status** Resolved

Schema IfcTypeDefResource **Version** R1.5 - Pre-Beta

Issue Description IfcTypeDefResource - IfcPropertyTypeDef

Currently SharedProperties is mandatory, but we have type def's that define only occurrence properties

Proposed Solution make optional

Resolution Agreed. Corrected by TL.

Issue Number **I - 209** **Issue Date** 8/23/97

Author Liebich **Owner** Liebich **Status** Resolved

Schema IfcTypeDefResource **Version** R1.5 - Pre-Beta

Issue Description IfcTypeDefResource - IfcSimpleProperty

Currently ValueComponent is OPTIONAL, but it should be always given

Proposed Solution Make mandatory

Resolution Agreed. Corrected by TL.

Issue Number **I - 210** **Issue Date** 8/23/97

Author Liebich **Owner** Liebich **Status** Resolved

Schema IfcTypeDefResource **Version** R1.5 - Pre-Beta

Issue Description Now that that other predefined properties have been combined into this schema, the name TypeDefResource no longer seems appropriate.

Proposed Solution Rename this schema to "IfcPropertyTypeResource"

Resolution Agreed

Action # 1 **Assignee** Liebich **Status** Complete **Resolved in Version** R1.5 - Final

make the change as proposed.

Issue Number **I - 211** **Issue Date** 8/23/97

Author Liebich **Owner** See **Status** Rejected

Schema IfcModelingAidExt **Version** R1.5 - Pre-Beta

Issue Description ModelingAid entities don't 'feel' like Core Layer concepts. They 'feel' more like resources. If PlacementRelToGrid is generalized to "ConstrainedPlacement"s, then it should be possible to push all of the ModelingAid entities down to the resource layer.

Proposed Solution Push all of the ModelingAid entities down to the resource layer.

Resolution Rejected - This is a problem with regard to placement of model elements. Presumably, "ConstrainedPlacement" should be subtyped from LocalPlacement. LocalPlacement references an IfcObject (as the 'relative to') object. This would mean that you could not place elements relative to ModelingAids.

Issue Number **I - 212** **Issue Date** 9/5/97

Author See **Owner** Liebich **Status** Resolved

Schema IfcKernel ***Version*** R1.5 - Pre-Final

Issue Description LocalPlacement.PlacementRelTo [IfcObject] - placement relative to an IfcObject is a problem -- many IfcObjects done have geometry and therefore don't have a placement that can be used (relative to)

Proposed Solution There are really only two subtypes that have placement (that can be referenced) - IfcProduct and IfcModelingAid. Please add a WHERE rule limiting to these OR create a select type which is referenced by LocalPlacement.

Resolution This should be done with a SelectType called "IfcObjectWithPlacement"

Action # 1 ***Assignee*** Liebich ***Status*** Complete ***Resolved in Version*** R1.5 - Final

make the change as resolved.

Issue Number I - 213 ***Issue Date*** 9/8/97

Author Liebich ***Owner*** Liebich ***Status*** Resolved

Schema IfcProductExt ***Version*** R1.5 - Pre-Final

Issue Description IfcSiteComplex and IfcBuildingComplex: Both do not define any particular data, they just carry the meaning that this group only contains either sites or buildings.

Proposed Solution Delete both classes, use the direct instanciation of IfcGroup instead, and make use of the new GroupPurpose attribute to indicate an SiteComplex or a BuildingComplex. Add this to documentation.

Resolution Agreed. Eliminate these two classes and document the use of IfcGroup with the "GroupPurpose" set to SiteComplex and BuildingComplex, respectively

Action # 1 ***Assignee*** Liebich ***Status*** Complete ***Resolved in Version*** R1.5 - Final

Eliminate the classes.

Action # 2 ***Assignee*** Liebich ***Status*** Incomplete ***Resolved in Version*** R1.5 - Final

Update the documentation for IfcGroup to describe its use for this purpose.

Not complete as of 26-Nov-97.

Issue Number I - 214 ***Issue Date*** 9/18/97

Author See ***Owner*** See ***Status*** Deferred to R2.0

Schema IfcControlExt ***Version*** R1.5 - Pre-Final

Issue Description We need a general purpose constraint mechanism to support code checking constraints in particular, but can also be used for things like designer imposed constraints.

Proposed Solution See general purpose constraint proposal from the CS-1 team -- would like to see this introduced in R1.5 so that it can be used to develop solutions for CS-1 and CS-2 projects in R2.0

Resolution Agreed.

Action # 1 ***Assignee*** See ***Status*** Incomplete ***Resolved in Version*** R2.0 - Beta

Introduce IfcControlExt schema including general purpose constraint as agreed with STF and CS teams.

Action # 2 ***Assignee*** Liebich ***Status*** Incomplete ***Resolved in Version*** R2.0 - Beta

Review the general purpose constraint mechanism proposed by the CS team and make comments

Action # 3 ***Assignee*** Wix ***Status*** Incomplete ***Resolved in Version*** R2.0 - Beta

Review the general purpose constraint mechanism proposed by the CS team and make comments

Action # 4 ***Assignee*** Forester ***Status*** Incomplete ***Resolved in Version*** R2.0 - Beta

Review the general purpose constraint mechanism proposed by the CS team and make comments

---

Issue Number I - 215 ***Issue Date*** 9/18/97

Author	See	***Owner***	Liebich	***Status***	Resolved
Schema	IfcUtilityResource	***Version***	R1.5 - Pre-Final		

Issue Description Class: IfcAuditTrail -- Attribute "AuditTrailLength" - which holds the length of the Audit trail length was agreed, but is still not in. We did agree that we would limit this to a single transaction, (through where rules limits), but this attribute is needed to insure backward compatibility in future versions.

Proposed Solution Add this attribute (type integer). NOTE: this allows an owning application to "set" the length for this trail on an object by object basis.

Resolution Agreed

Action # 1 ***Assignee*** Liebich ***Status*** Complete ***Resolved in Version*** R1.5 - Final

Make the change as proposed

(RS) 26-Nov-97: in the .HTML, the data type, min, max, default not set for "AuditTrailLength"

Issue Number I - 216 ***Issue Date*** 9/18/97

Author	See	***Owner***	Liebich	***Status***	Resolved
Schema	IfcUtilityResource	***Version***	R1.5 - Pre-Final		

Issue Description Class: IfcAuditTrail -- Transactions [IfcTransaction] -- cardinality should be limited to the AuditTrialLength (discussed above).

Proposed Solution change cardinality to List [0:AuditTrailLength]. Note: this assumes IfcTransaction will be contained within IfcAuditTrail and will be made a 'friend' to the IfcAuditTrail.

Resolution [TL] to I-215, I-216: The final chosen resolution is adding a WHERE clause
WR1: HIINDEX(Transactions) <= 1;
[RS] No -- agreed compromise was to make this attribute derived so that it is available for query
[RS] No -- this is STILL not right. The original intention was for the owning application to have control of the length of this trail. Therefore, it should not be derived, but set.

Action # 1 ***Assignee*** Liebich ***Status*** Complete ***Resolved in Version*** R1.5 - Final

Add the attribute

Issue Number I - 217 ***Issue Date*** 9/18/97

Author	See	***Owner***	Liebich	***Status***	Rejected
Schema	IfcUtilityResource	***Version***	R1.5 - Pre-Final		

Issue Description Class: IfcRegisteredApplication -- ApplicationDeveloper [ref [IfcActor]] -- shouldn't we make this an integer index into the TeamRegistry as in other places?

Proposed Solution change data type to INTEGER and document that this is index into ProjectTeam.

Resolution [TL] disagreed: the semantic of TeamRegistry is to register team members of the AEC project. I don't see, that an application developer becomes a member of the Project Team. Recommendation: leave it as it is.
[RS] agreed

Issue Number I - 218 ***Issue Date*** 9/18/97

Author	See	***Owner***	Liebich	***Status***	Resolved
Schema	IfcUtilityResource	***Version***	R1.5 - Pre-Final		

Issue Description IfcTable -- revised schema will allow multiple headings (only one of which will be used).

Proposed Solution [TL] There are multiple headings, look at diagram in MS word document "RAS_R15rev4_Compsite_1d.doc"

[RS] Yes, this is valid. However, you need to establish a convention for interpreting where the headings span multiple columns (e.g. if heading for col, 3,4,5 are blank, then col 2 heading extends for all for columns).

Resolution Leave schema as it is, but add documentation to clarify convention for interpreting where the headings span multiple columns.
[RS] This still leaves a problem. Currently, the NumberOfRows will include both the data rows

and the heading rows. How will one query for the number of data rows?
Final Resolution: 1) class definition modified so that Rows is LIST [0:?], NumberOfDataRows and NumberOfHeadingRows are now derived attributes.

Action # 1 **Assignee** Liebich **Status** Complete **Resolved in Version** R1.5 - Final

complete items 1 & 2

Issue Number **I - 219** **Issue Date** 9/18/97

Author See **Owner** Liebich **Status** Resolved

Schema IfcUtilityResource **Version** R1.5 - Pre-Final

Issue Description Class: IfcTable -- NR, NC -- These names are awfully cryptic.

Proposed Solution Change them back to NumberOfRows and NumberOfColumns (as before).

Resolution Agreed

Action # 1 **Assignee** Liebich **Status** Complete **Resolved in Version** R1.5 - Final

make the change

Issue Number **I - 220** **Issue Date** 9/18/97

Author See **Owner** Liebich **Status** Resolved

Schema IfcUtilityResource **Version** R1.5 - Pre-Final

Issue Description Class: IfcTable -- NR, NC -- If the Rows and RowValue lists were made 0:?, then these values could (and should) be derived.

Proposed Solution Change lists to 0:? And make these attributes derived.

Resolution Agreed

Action # 1 **Assignee** Liebich **Status** Complete **Resolved in Version** R1.5 - Final

Make the change as proposed

Issue Number **I - 221** **Issue Date** 9/18/97

Author See **Owner** Liebich **Status** Resolved

Schema IfcUtilityResource **Version** R1.5 - Pre-Final

Issue Description Class: IfcTable -- Rows [List[1:NR] of IfcTableRow] -- This will result in one too few rows unless NR is defined to be the number of rows + 1 (for the headings).

Proposed Solution Change cardinality to List[1:NR+1]

Resolution [TL] why not considering a heading just as another row?
[RS] agreed so long as the documentation is clear that headings are inlcuded. However, this kind of defeats the purpose of the values for NumberOfRows and NumberOfColumns (since you won't really know how many data rows you have until you check to see which ones are headings.
Final resolution: will reverse the direction of the relationship to TableRows and will change the attribute "Rows" to 2 attributes (both derived values) - for "NumberOfDataRows" and "NumberOfHeadingRows".

Action # 1 **Assignee** Liebich **Status** Complete **Resolved in Version** R1.5 - Final

make changes as resovled.

Issue Number **I - 222** **Issue Date** 9/18/97

Author See **Owner** Liebich **Status** Resolved

Schema IfcUtilityResource **Version** R1.5 - Pre-Final

Issue Description Class: IfcTableRow -- RowValues [List[1:NC] IfcMeasureValue] -- this still violates encapsulization. Also, values should be contained and not "Ref" erenced as they are now.

Proposed Solution 1) move renamed NC to IfcTableRow class (still derived) since is is only needed in this contained object.

2) change "Ref IfcMeasureValue" to just "IfcMeasureValue" --> "List[1:NoOfColumns] IfcMeasureValue"

Resolution Agreed.

Action # 1 **Assignee** Liebich **Status** Complete **Resolved in Version** R1.5 - Final

Make the change as proposed.

(RS) 26-Nov-97: "NumberOfColumns" not moved into IfcTableRow yet - note: "List[1:NoOfColumns]" defined at data type for "RowValues" violates encapsulization. Also, values should be contained and not "Ref" erenced as they are now.

Issue Number I - 223 **Issue Date** 9/18/97

Author See **Owner** Liebich **Status** Rejected

Schema IfcGeometryResource **Version** R1.5 - Pre-Final

Issue Description Type: IfcAxis2Placement -- Naming convention recommendation.

Proposed Solution All Select types should be called "IfcXxxxSelect".

Resolution [TL] agreed in general, but disagreed in particular: one modeling principle in Pewsey was to leave STEP names as they are

[RS] Not agreed. We have already renamed the class names. What is the problem with being consistent with the names of Select types too?

Action # 1 **Assignee** Liebich **Status** Eliminated **Resolved in Version** R1.5 - Final

Review all schemata to insure that all Select types follow the naming convention.

Issue rejected after all --

Issue Number I - 224 **Issue Date** 9/18/97

Author Liebich **Owner** Liebich **Status** Resolved

Schema IfcGeometryResource **Version** R1.5 - Pre-Final

Issue Description We don't have a Point entity presently

Proposed Solution add the class "IfcPoint" for backward compatibility from R2.0 -- in the gray page Network we already use another subtype IfcPointOnCurve

Resolution Agreed

Action # 1 **Assignee** Liebich **Status** Complete **Resolved in Version** R1.5 - Final

Add the class - coordinated with STEP P42

Issue Number I - 225 **Issue Date** 9/18/97

Author Liebich **Owner** Liebich **Status** Resolved

Schema IfcGeometryResource **Version** R1.5 - Pre-Beta

Issue Description Class IfcCurveBoundedSurface -- name clashes with STEP entity curve_bounded_surface

Proposed Solution Rename into IfcCurveBoundedPlane, this is more precise. Change data type of BasisSurface to IfcPlane to be more precise

Resolution Agreed

Action # 1 **Assignee** Liebich **Status** Complete **Resolved in Version** R1.5 - Final

make the changes described

Issue Number I - 226 **Issue Date** 9/18/97

Author Liebich **Owner** Liebich **Status** Resolved

Schema IfcGeometryResource **Version** R1.5 - Pre-Final

Issue Description We don't have a fundamental "Surface" supertype. This will be needed in R2.0 for the HVAC model (IfcCylindricalSurface)

Proposed Solution Add the class "IfcElementarySurface" for upward compatibility with R2.0 -- basis non planar surfaces such as IfcCylindricalSurface.

Resolution Agreed.

Action # 1 **Assignee** Liebich **Status** Complete **Resolved in Version** R1.5 - Final

Add the class - compatible with P42

Issue Number **I - 227** **Issue Date** 9/18/97

Author Liebich **Owner** Liebich **Status** Resolved

Schema **Version** R1.5 - Pre-Final

Issue Description We don't have a fundamental base type for non-closed Breps. This will be needed for the HVAC model in R2.0.

Proposed Solution Add the class "IfcConnectedFaceSet" for upward compatibility -- supertype for non closed Breps

Resolution Agreed

Action # 1 **Assignee** Liebich **Status** Complete **Resolved in Version** R1.5 - Final

Add the class - compatible with P42

Issue Number **I - 228** **Issue Date** 9/18/97

Author Liebich **Owner** Liebich **Status** Resolved

Schema IfcGeometryResource **Version** R1.5 - Pre-Final

Issue Description Class IfcAttDrivenExtrusionSolid -- This name is not consistent with others

Proposed Solution 1) Rename into IfcAttDrivenExtrudedSolid for naming consistency with IfcExtrudedAreaSolid.

2) Group List of IfcExtrusionSegment and List of Path Length (corresponding Lists) into a single List of IfcAttDrivenExtrudedSegment. Note: this was an implementers request at the Munich meeting.

Resolution Agreed

Action # 1 **Assignee** Liebich **Status** Complete **Resolved in Version** R1.5 - Final

Make the changes as proposed

Issue Number **I - 229** **Issue Date** 9/18/97

Author Liebich **Owner** Liebich **Status** Resolved

Schema IfcGeometryResource **Version** R1.5 - Pre-Final

Issue Description Class IfcAttDrivenExtrusionSolid -- We don't have the baseline for these entities defined

Proposed Solution Add (DER) Path, defines the ExtrudedSolid "Baseline" to which we relate the material layer set base line. It is computed by the function IfcExtrusionPath

Resolution [RS] agreed, however, determining the path indirectly is a bit troubling. See also comments on attachment of MaterialLayerSets too high in the model.

Action # 1 **Assignee** Liebich **Status** Complete **Resolved in Version** R1.5 - Final

make the changes as described

Issue Number **I - 230** **Issue Date** 9/18/97

Author Liebich **Owner** Liebich **Status** Resolved

Schema IfcGeometryResource **Version** R1.5 - Pre-Final

Issue Description Class IfcAttDrivenExtrusionSolid -- simplification of ExtrudedSolid segments means that the "position" (placement) should be moved back up to this class.

Proposed Solution 1) Add position back to IfcAttDrivenExtrudedSolid, since it now defines the path as well.
2) Eliminate the IfcStraightPathDef.

[RS] StraightPathDef is now default in the revised (now concrete) IfcAttDrivenExtrudedSolid

Resolution Is there some disagreement about StraightPathDef?

IfcStraightPathDef is omited but information is present in IfcAttDrivenExtrudedSolid and IfcAttDrivenExtrudedSegment.

Issue Number	***I - 231***			***Issue Date***	9/18/97
Author	Liebich	***Owner***	Liebich	***Status***	Resolved
Schema	IfcGeometryResource	***Version***	R1.5 - Pre-Final		

Issue Description We don't currently have a class for Att Driven revolved solids. Additionally, the design for a series of extrusion segments could be improved with the concept of an extrusion segment (Note: this was an implementers request at the Munich meeting.)

Proposed Solution 1) Add IfcAttDrivenRevolvedSolid,
2) eliminate the IfcArcPathDef,
3) Group List of IfcExtrusionSegment and List of Path Length (corresponding Lists) into a single List of IfcAttDrivenExtrudedSegment.

Resolution Agreed with resolution of the following question -- Do we have some confusion about the ArcPathDef? Are all paths now a simple curve (line or arc) ?

Yes, simplification from R1.0 to R1.5 was to delay support for polycurve paths to some future release. R1.5 supports the straight path that can be derived from an AttDrivenExtrudedSolid defintion and the arc path that can be derived from an AttDrivenRevolvedSolid.

Action #	1	***Assignee***	Liebich	***Status***	Complete	***Resolved in Version***	R1.5 - Final

make the changes as described

Issue Number	***I - 232***			***Issue Date***	9/18/97
Author	Liebich	***Owner***	Liebich	***Status***	Resolved
Schema	IfcGeometryResource	***Version***	R1.5 - Pre-Final		

Issue Description Class IfcAttDrivenRevolvedSolid -- we need to insure that we can place material layers for such a solid.

Proposed Solution Add (DER) Path, defines the baseline for the extrusion, to which we relate the material layer set baseline. It is computed by the function IfcRevolutionPath.

Resolution [RS] agreed, however, determining the path indirectly is a bit troubling. See also comments on attachment of MaterialLayerSets too high in the model.

Action #	1	***Assignee***	Liebich	***Status***	Complete	***Resolved in Version***	R1.5 - Final

make the changes as described

Issue Number	***I - 233***			***Issue Date***	9/18/97
Author	Liebich	***Owner***	Liebich	***Status***	Resolved
Schema	IfcGeometryResource	***Version***	R1.5 - Pre-Final		

Issue Description Class IfcAttDrivenRevolvedSolid -- Now that extrusion segments are self contained and dependent on the placement of the parent ExtrudedSolid, we need the "position" (placement) back in this class.

Proposed Solution Add position back to IfcAttDrivenRevolvedSolid, since it now defines the path as well.

Resolution Agreed. -- but later superseded by other changes in the definition of AttDrivenExtrusionSegments. Placement was finally added for each of the Segments (see I292)

Action #	1	***Assignee***	Liebich	***Status***	Eliminated	***Resolved in Version***	R1.5 - Final

make the change as described.

Issue Number	***I - 234***			***Issue Date***	9/18/97
Author	Liebich	***Owner***	Liebich	***Status***	Resolved
Schema	IfcGeometryResource	***Version***	R1.5 - Pre-Final		

Issue Description Class IfcExtrusionSegment -- this name is not inconsistent. It should also be moved under IfcExtrudedAreaSolid (Note: this was a request from the implementers meeting in Munich).

Proposed Solution Rename into IfcAttDrivenExtrudedSegment for naming consistency. Now subtyped from IfcExtrudedAreaSolid. The explicit attributes are overridden by Derived Attributes, since it is driven by those attributes.

Resolution Agreed

Action #	1	***Assignee***	Liebich	***Status***	Complete	***Resolved in Version***	R1.5 - Final

Make the change as described

Issue Number	***I - 235***			***Issue Date***	9/18/97
Author	Liebich	***Owner***	Liebich	***Status***	Resolved
Schema	IfcGeometryResource	***Version***	R1.5 - Pre-Final		

Issue Description Now that we have added a RevolvedSolid, we will need segments.

Proposed Solution Add the new class "IfcAttDrivenRevolvedSegment" for revolved segments, it is subtyped from IfcRevolvedAreaSolid, since both define the same functionality. The explicit attributes are overridden by Derived Attributes, since it is driven by those attributes.

Resolution Agreed

Action #	1	***Assignee***	Liebich	***Status***	Complete	***Resolved in Version***	R1.5 - Final

make the addition as described

Issue Number	***I - 236***			***Issue Date***	9/18/97
Author	Liebich	***Owner***	Liebich	***Status***	Rejected
Schema	IfcGeometryResource	***Version***	R1.5 - Pre-Final		

Issue Description Class IfcTaperedExtrusionSegment -- This class could not be used for sloped walls as discussed in Sep-97 Munich implementers meeting. Also, it was pointed out that resulting shaped _could_ be defined using morphed extrusions.
Is it to specialized? Should we reduce class count?

Proposed Solution Consider deleting this class.

Resolution Not agreed. Leave it in as a convenient way to do uniformly tapered shapes.

Issue Number	***I - 237***			***Issue Date***	9/18/97
Author	Liebich	***Owner***	Liebich	***Status***	Resolved
Schema	IfcGeometryResource	***Version***	R1.5 - Pre-Final		

Issue Description Class IfcMorphingExtrusionSegment -- name is inconsistent with new scheme. It is also possible to define morphed segments that twist.

Proposed Solution Rename to IfcAttDrivenMorphedExtrudedSegment for naming consistency. Add a where rule that requires the start and end profile to have the same orientation (to avoid twisted configurations)

Resolution [RS] agreed. However, note that about all you can do is insure that the LCS does not rotate between profile 'A' and 'B', this does not insure that the user/programmer did not rotate the profile within the second LCS.

Action #	1	***Assignee***	Liebich	***Status***	Complete	***Resolved in Version***	R1.5 - Final

make the changes as described

Issue Number	***I - 238***			***Issue Date***	9/18/97
Author	Liebich	***Owner***	Liebich	***Status***	Resolved

Schema IfcGeometryResource ***Version*** R1.5 - Pre-Final

Issue Description Need to add segment object for revolved extrusions (new) that morph.

Proposed Solution Introduce a new class for morphed revolved segments -- "IfcAttDrivenMorphedRevolvedSegment", using the same constraints as for IfcAttDrivenMorphedExtrudedSegment

Resolution [RS] agreed. Perfect example for the graphics on this is a curved spread footing wall where the wall slopes.

Action # 1 ***Assignee*** Liebich ***Status*** Complete ***Resolved in Version*** R1.5 - Final

add new class and example of use in documentation

Action # 2 ***Assignee*** Liebich ***Status*** Complete ***Resolved in Version*** R2.0 - Alpha

Add an example diagram for morphing, revolved segment.

Issue Number I - 239 ***Issue Date*** 9/18/97

Author See ***Owner*** Liebich ***Status*** Resolved

Schema IfcGeometryResource ***Version*** R1.5 - Pre-Final

Issue Description Classes: IfcAttDrivenProfileDef, IfcArbitraryProfileDef -- CurveForSurface [IfcBoundedCurve] -- In Implementers meeting (9-Sep), we discussed moving this down to the ArbitraryProfileDef level and thus eliminate all of the DER redefinings in the other subtypes.

Proposed Solution Move this attribute down to IfcArbitraryProfileDef

Resolution Agreed

Action # 1 ***Assignee*** Liebich ***Status*** Complete ***Resolved in Version*** R1.5 - Final

make the change as proposed

Issue Number I - 240 ***Issue Date*** 9/18/97

Author See ***Owner*** Liebich ***Status*** Resolved

Schema IfcPropertyTypeResource ***Version*** R1.5 - Pre-Final

Issue Description Class: IfcPropertyTypeDef -- Agreed attribute for identifying the domain point of view from which a 'type' is defined (from Pewsey) -- is missing.

Proposed Solution Attribute called "ObjTypeDomainView" [IfcObjTypeViewpointsEnum].

Resolution Agreed

Action # 1 ***Assignee*** Liebich ***Status*** Complete ***Resolved in Version*** R1.5 - Final

make the change as proposed.

Issue Number I - 241 ***Issue Date*** 9/18/97

Author See ***Owner*** Liebich ***Status*** Resolved

Schema IfcPropertyTypeResource ***Version*** R1.5 - Pre-Final

Issue Description Class: IfcPropertyTypeDef -- TypeReference [IfcPropertyTypeDef] -- We need to establish a convention for the way references to other TypeDefs will be done.

Proposed Solution 1) Establish the convention that ALL references to other TypeDefs (in the subject TypeDef) is to the parent TypeDef. Example: TypeDef for the Specific WindowType "WoodFrameAwning" references TypeDef "Awning", which references TypeDef "Window". 2) rename the attribute to "ParentTypeDef"

Resolution [TL] agreed and done as ParentTypeReference - (INV) ReferencedByChildType
[RS] Good! This will be used by the new definitions for Door and Window property sets.

Action # 1 ***Assignee*** Liebich ***Status*** Complete ***Resolved in Version*** R1.5 - Final

change as described

Issue Number	***I - 242***			***Issue Date***	9/18/97
Author	See	***Owner***	Liebich	***Status***	Rejected
Schema	IfcPropertyTypeResource	***Version***	R1.5 - Pre-Final		

Issue Description Class: IfcOccurrencePropertySet, IfcSharedPropertySet -- I am uncomfortable with the rational for introducing these two subtypes because they don't add anything.

Proposed Solution eliminate them.

Resolution [TL] still prefer to leave them in, since they utilize semantically different concepts and have different attributes
[RS] agreed in the spirit of cooperation.

Issue Number	***I - 243***			***Issue Date***	9/18/97
Author	See	***Owner***	Liebich	***Status***	Rejected
Schema	IfcPropertyTypeResource	***Version***	R1.5 - Pre-Final		

Issue Description Class: IfcRepresentationContext -- ProjectID [IfcProjectUniqueID] -- This isn't really needed. If we take the convention that objects from this class should be contained in the ShapeRepresentation.

Proposed Solution eliminate attribute.

Resolution [TL] disagreed: an instance of IfcRepresentationContext can be shared among multiple instances of IfcShapeRepresentation, it can therefore not be contained
[RS] Okay; agreed -- leave it as is.

Issue Number	***I - 244***			***Issue Date***	9/18/97
Author	See	***Owner***	Liebich	***Status***	Resolved
Schema	IfcPropertyTypeResource	***Version***	R1.5 - Pre-Final		

Issue Description Class: IfcProductShape -- RootComponent [IfcProductComponentShape] -- This attribute name is a bit uncomfortable in this it is really the resultant product shape (not the root).

Proposed Solution rename it to ProductShape

Resolution [TL] agreed

Action # 1 ***Assignee*** Liebich ***Status*** Complete ***Resolved in Version*** R1.5 - Final

make change as proposed

Issue Number	***I - 245***			***Issue Date***	9/18/97
Author	See	***Owner***	Liebich	***Status***	Resolved
Schema	IfcPropertyTypeResource	***Version***	R1.5 - Pre-Final		

Issue Description Class: IfcShapeBody -- AnalysisTag [STRING] -- If this is the descriptor for standardized components in product shapes (loose link to semantic model attributes side of model), then this name is misleading.

Proposed Solution 1) rename to StdComponentDescriptor, 2) pump up the documentation to insure that EVERYBODY understands the relationship between the StdComponentDescriptorsEnum (no the semantic model side) and use of them here on the shape models for each component. This is the only reliable way applications will have to know which parts of the geometry corresponde to known parts of products (e.g. a Window frame or glazing).

Resolution [TL] should be done as ComponentDescriptor::STRING, Note: we cannot use enum there, since then the resource would depend on lower level schemas - violation of IFC Architecture
[RS] NOTE: use of a STRING here is VERY weak. We MUST look for a stronger link between the semantic model attributes that must 'drive' the AttDrivenGeom. Thomas to look into doing this in R2.0 using Schema rules (? Can't remember the exact name)

Action # 1 ***Assignee*** Liebich ***Status*** Complete ***Resolved in Version*** R1.5 - Final

Add the ComponentDescriptor to the ComponentShapeRep

Action # 2 **Assignee** Liebich **Status** Incomplete **Resolved in Version** R3.0 - Beta

Develop method by which ComponentShapeReps will be 'driven' from attributes on the semantic model object to which the ShapeRep is related.

Issue Number I - 246 **Issue Date** 9/18/97

Author See **Owner** Liebich **Status** Deferred to R2.0

Schema IfcPropertyTypeResource **Version** R1.5 - Pre-Final

Issue Description Class: IfcShapeRepresentation -- UsageTag [STRING] -- Doc says that this is to identify usage for this shape (e.g contrours or boundaries for Site). This seems very WEAK at this point; expecially given that it is only a STRING. How will we achieve any consistency across vendors, let alone users.

Proposed Solution No proposal developed at this point.

Resolution It is acknowledged that UsageTag is weak and somewhat redundant with the RepresentationType already on the ShapeRep. However, we do not have a better solution in time for R1.5. Therefore, we are going to defer this for resolution in R2.0.

Action # 1 **Assignee** See **Status** Complete **Resolved in Version** R2.0 - Alpha

Add to the list of R2.0 STF projects

Issue Number I - 247 **Issue Date** 9/18/97

Author Liebich **Owner** Liebich **Status** Resolved

Schema IfcPropertyTypeResource **Version** R1.5 - Pre-Final

Issue Description Class IfcShapeResult -- in some cases, the shape result will be a standard component shape (e.g. a Window "Frame").

Proposed Solution add ComponentDescriptor, since also the result can be a standard component, referenced by a semantic type, e.g. the union of all four frame sides

Resolution Agreed.

Action # 1 **Assignee** Liebich **Status** Complete **Resolved in Version** R1.5 - Final

make the addition as proposed

Issue Number I - 248 **Issue Date** 9/18/97

Author See **Owner** Liebich **Status** Resolved

Schema IfcPropertyResource **Version** R1.5 - Pre-Final

Issue Description Class: IfcMaterialLayerSetUsage -- SenseLtoR [Boolean] -- naming convention dictates other name.

Proposed Solution rename to "MaterialLayersLtoR".

Resolution [TL] should be done as "MlsSenseLtoR", note we uses the abbrevation Mls everywhere else
[RS] agreed

Action # 1 **Assignee** Liebich **Status** Complete **Resolved in Version** R1.5 - Final

make the change as resolved.

Issue Number I - 249 **Issue Date** 9/18/97

Author See **Owner** Liebich **Status** Resolved

Schema IfcPropertyResource **Version** R1.5 - Pre-Final

Issue Description Class: IfcMaterialLayerSetUsage -- CenterOffset [IfcLengthMeasure] -- This is the old naming and method. Additionally, this attribute is not needed as it is redundant with the one discussed next.

Proposed Solution remove the attribute.

Resolution Agreed

Action # 1 **Assignee** Liebich **Status** Complete **Resolved in Version** R1.5 - Final

change as proposed

Issue Number **I - 250** **Issue Date** 9/18/97

Author See **Owner** Liebich **Status** Resolved

Schema IfcPropertyResource **Version** R1.5 - Pre-Final

Issue Description Class: IfcMaterialLayerSetUsage -- CenterOffsetFromPath [IfcLengthMeasure] -- This is the old naming and method.

Proposed Solution Rename to MlsBaselineOffset.

Resolution [TL] will be done to comply with drawing from May STF mtg (done by JF)

Action # 1 **Assignee** Liebich **Status** Complete **Resolved in Version** R1.5 - Final

change as resolved.

Issue Number **I - 251** **Issue Date** 9/18/97

Author See **Owner** Liebich **Status** Resolved

Schema IfcPropertyResource **Version** R1.5 - Pre-Final

Issue Description Class: IfcMaterialLayerSetUsage -- TotalWidth [IfcLengthMeasure] -- I made the case in the last set of comments (and believe we agreed in Pewsey) that this dimension is virtually all cases is better referred to as the "thickness".

Proposed Solution rename to "TotalThickness".

Resolution [TL] should be "MlsTotalThickness" according to the diagram. Also, the function IfcMlsTotalThickness must be updated to new layer definition.

[RS] main point here was the use of the term "Thickness" instead of "Width"

Action # 1 **Assignee** Liebich **Status** Complete **Resolved in Version** R1.5 - Final

change as resolved.

Issue Number **I - 252** **Issue Date** 9/18/97

Author See **Owner** Liebich **Status** Resolved

Schema IfcPropertyResource **Version** R1.5 - Pre-Final

Issue Description We need to be able to reference objects (other than simple property objects) from with PropertySets. For example, to specify a IfcDocument from within a PropertySet -- say for a cost estimate or construction schedule.

Proposed Solution wrap a ProjectUniqueID in a subtype of IfcProperty so that such references (essentially object pointers) can be included in PropertySets. Call the new property subtype "IfcObjectReference "

Resolution Agreed

Action # 1 **Assignee** Liebich **Status** Complete **Resolved in Version** R1.5 - Final

Add the "IfcObjectReference " subtype of IfcProperty in the IfcPropertyTypeResource

---

Issue Number **I - 253** **Issue Date** 9/18/97

Author See **Owner** Liebich **Status** Resolved

Schema IfcPropertyResource **Version** R1.5 - Pre-Final

Issue Description Class: IfcCoordinatedUniversalTimeOffset -- Ahead [IfcAheadOrBehind] -- it was agreed in Pewsey that this should be a Boolean, so why introduce the intermediate type?

Proposed Solution Eliminate IfcAheadOrBehind and make "Ahead" a Boolean.

Resolution Agreed

Action #	1	***Assignee***	Liebich	***Status***	Complete	***Resolved in Version***	R1.5 - Final
		change as resolved					

Issue Number ***I - 254*** ***Issue Date*** 9/18/97

Author	See	***Owner***	Liebich	***Status***	Resolved
Schema	IfcKernel	***Version***	R1.5 - Pre-Final		

Issue Description Class: IfcModelingAid -- The IR log from Pewsey says that this should be subtyped from IfcRoot, not IfcObject.

Proposed Solution Subtype from IfcRoot.

Resolution Agreed

Action #	1	***Assignee***	Liebich	***Status***	Complete	***Resolved in Version***	R1.5 - Final
		change as proposed					

Issue Number ***I - 255*** ***Issue Date*** 9/18/97

Author	See	***Owner***	Liebich	***Status***	Resolved
Schema	IfcKernel	***Version***	R1.5 - Pre-Final		

Issue Description Class: IfcLocalPlacement -- In our discussions in Munich (10-Sep-97, we agreed that IfcLocalPlacement should be subtyped from IfcModelingAid.

Proposed Solution Subtype from IfcModelingAid.

Resolution Agreed

Action #	1	***Assignee***	Liebich	***Status***	Complete	***Resolved in Version***	R1.5 - Final
		change as proposed.					

Issue Number ***I - 256*** ***Issue Date*** 9/18/97

Author	See	***Owner***	Liebich	***Status***	Resolved
Schema	IfcKernel	***Version***	R1.5 - Pre-Final		

Issue Description Class: IfcObject -- TypeDefinition List [0:?] [IfcPropertyTypeDef] -- Convention has been use plural naming for attributes with such cardinality.

Proposed Solution Rename to "TypeDefinitions".

Resolution Agreed

Action #	1	***Assignee***	Liebich	***Status***	Complete	***Resolved in Version***	R1.5 - Final
		change as proposed.					

Issue Number ***I - 257*** ***Issue Date*** 9/18/97

Author	See	***Owner***	Liebich	***Status***	Rejected
Schema	IfcKernel	***Version***	R1.5 - Pre-Final		

Issue Description Class: IfcProduct -- ProductShape [IfcProductShape] -- shouldn't this be a List ? For example, one to hold the BoundingBox rep, another to hold the AttDrivenShape rep and a third to hold the Explicit Shape rep.

Proposed Solution Make it a list? Am I missing something?

Resolution [TL] the definition is different, you shall use many IfcShapeRepresentation instead, each is characterized by the RepresentationType as either BoundingBox, AttributeDriven or Explicit
[RS] agreed -- no change needed

Issue Number ***I - 258*** ***Issue Date*** 9/18/97

Author	See	***Owner***	Liebich	***Status***	Resolved

Schema IfcKernel **Version** R1.5 - Pre-Final

Issue Description Class: IfcRelSequence -- INV IsPredecessorFrom S[0:?] -- should read "IsPredecessorTo".

Proposed Solution Rename to "IsPredecessorTo".

Resolution Agreed

Action # 1 **Assignee** Liebich **Status** Complete **Resolved in Version** R1.5 - Final

change as proposed.

Issue Number I - 259 **Issue Date** 9/18/97

Author See **Owner** Liebich **Status** Resolved

Schema IfcKernel **Version** R1.5 - Pre-Final

Issue Description Class: IfcRelSequence -- Cardinality on these relationships reads as 1to1 on the primary rels and NtoN in the Inverse rels

Proposed Solution Reset so that it is truly 1toN, one predecessor to many successors. Note: as discussed in Pewsey, some relationships are truly NtoN (as with this one). Documentation should be clear that, in these cases, it is necessary to create multiple relationships where there are multiple predecessors to a WorkTask.

Resolution Agreed.

Changed after I-200 in which KY argued that IfcSequence should be a subtype of IfcRelationship1to1 in all cases. Therefore this issue has been superseded.

Action # 1 **Assignee** Liebich **Status** Eliminated **Resolved in Version** R1.5 - Final

Correct cardinality as proposed.

Action # 2 **Assignee** Liebich **Status** Complete **Resolved in Version** R1.5 - Final

Insure that documentation is clear about the need for applications to create multiple relationships where relationships are truly NtoN (as the model now only supports 1to1 relationships).

(RS) 26-Nov-97: not done in Final-Candidate HTML reference docs.

Issue Number I - 260 **Issue Date** 9/18/97

Author See **Owner** Liebich **Status** Resolved

Schema IfcKernel **Version** R1.5 - Pre-Final

Issue Description Class: IfcLocalPlacement -- This class was moved to ModelAidExtension.

Proposed Solution Remove it from the Kernel. Reference should also be removed from diagram 3.

Resolution [TL] disagreed and error found: moving IfcLocalPlacement down to IfcModelingAid would cause a violation of the IFC Architecture, since IfcProduct.LocalPlacement is using IfcLocalPlacement and would now reference a schema on a higher level. Recommendation: leave it in IfcKernel
[RS] Agreed. TL will move LocalPlacement back into the Kernel (still subtyped from IfcModelingAid) and RS will remove and reference it from the ModelingAidExtension.

Action # 1 **Assignee** Liebich **Status** Complete **Resolved in Version** R1.5 - Final

Move localPlacement back to Kernel

Action # 2 **Assignee** See **Status** Complete **Resolved in Version** R1.5 - Final

Remove LocalPlacement from IfcModelingAidExtension and reference it there - from Kernel

---

Issue Number I - 261 **Issue Date** 9/18/97

Author See **Owner** See **Status** Resolved

Schema IfcProductExt **Version** R1.5 - Pre-Final

Issue Description Class: IfcBuildingElement -- HasMaterial [IfcMaterialSelect] -- this reference to materials is MUCH TOO HIGH in the model. Such references should be made at the leaf nodes, in the definition of

TypeDefs.

Proposed Solution Remove from BuildingElement and establish a convention for references to Materials, MaterialsLayersSets, etc. in TypeDefs.

Resolution [TL] disagreed: we have never seriously attempted to look at all consequences, when dealing with materials in Property Sets, in particular the connectivity problem, where we need material information, is required in ACS, but Type Definition and Property Sets are currently not in Exchange Class -- would have severe implications: Recommendation: leave Material as explicitly handled attribute for now and defer the issue to Release 2.0
[RS] The point is that one does not know how to specify materials until the detailed type is known. The type and configuration of materials is 'type driven'. Further, other attributes, which relate to material will be in Type Driven PropertySets. Therefore, references to Material should be done at the leaf node level -- in the Type Driven PropertySets. This will still be compatible with the Layer Priority scheme included in the IfcRelConnectsElements.
Final Resolution: 1) A new type of Materials reference will be added to the IfcPropertyResource -- for list of materials (IfcMaterialList). This will be referenced for things that have more than one material, but not arranged as MaterialLayers. 2) IfcMaterialSelect will now include IfcMaterialList and NOT include IfcMaterial. 3) documentation for subtypes of BuildingElement will be expanded to note which of the materialSelect types should be used (e.g. MaterialLayer for Walls, MaterialList for Windows and Doors). 4) references to materials in the Psets will reference one of the materials in these lists as an index in the list (e.g. a window frame Pset may reference material 3 in the list).

Action # 1 **Assignee** Liebich **Status** Complete **Resolved in Version** R1.5 - Final

TL to complete items 1, 2, 3, RS to complete item 4

(RS) 26-Nov-97: IfcMaterialSelect must not include IfcMaterial or else the use of indicies to reference materials (from Psets) will not work! Use an IfcMaterialList with a single material in those cases and eliminate IfcMaterial from IfcMaterialSelect

Action # 2 **Assignee** See **Status** Complete **Resolved in Version** R1.5 - Final

RS to complete item 4 described in the final resolution

Issue Number I - 262 **Issue Date** 9/18/97

Author See **Owner** Liebich **Status** Resolved

Schema IfcProductExt **Version** R1.5 - Pre-Final

Issue Description Class: IfcBuildingSection -- As discussed in Pewsey, if this class remains, it should allow type definition.

Proposed Solution Add the attribute "GenericType" of type IfcBldgSectionTypeEnum.

Resolution [TL] how does the IfcBldgSectionTypeEnum differ for IfcBuildingTypeEnum? Attaching another GenericType at IfcBuildingSection is impossible, since it inherits GenericType from superclass.
[RS] Cannot TypeDef BuildingSection because it is subtyped from Building, which already has a Type and EXPRESS will not let us override this. These EXPRESS limitations are a real pain sometimes! We should eliminate BuildingSection or define it such that it is not subtyped from Building.
Final resolutions: remove this class and include in the documentation the use of IfcZone to represent BuildingSections --

Action # 1 **Assignee** Liebich **Status** Complete **Resolved in Version** R1.5 - Final

remove the BuildingSection class

Action # 2 **Assignee** Liebich **Status** Complete **Resolved in Version** R1.5 - Final

Add to the IfcZone documentation about how to represent BuildingSections using Zones.

(RS) 26-Nov-97: Not done in Final-Candidate HTML reference docs.

Issue Number I - 263 **Issue Date** 9/18/97

Author See **Owner** Liebich **Status** Resolved

Schema IfcProductExt **Version** R1.5 - Pre-Final

Issue Description Class: IfcSpace -- As discussed in Pewsey, we need an average height for a space.

Proposed Solution Add and attribute "calc_AvgHeight" of type IfcPositiveLengthMeasure

Resolution [TL] agreed

Action # 1 **Assignee** Liebich **Status** Complete **Resolved in Version** R1.5 - Final

add attribute as resolved.

Issue Number **I - 264** **Issue Date** 9/18/97

Author	See	**Owner**	Liebich	**Status**	Resolved
Schema	IfcProductExt	**Version**	R1.5 - Pre-Final		

Issue Description Class: IfcRelConnectsElements -- there are 4 new attributes which are related to resolving drawing at connections of multilayered elements. This seems too specific for such a generalized class.

Proposed Solution Subtype a logical connector for objects using multiple layers and move these attributes to the subtype.

Resolution [TL] attributed attached as required by implementers, they are just INTEGER, and should therefore not create a big overhead
[RS] The point is that they don't make sense in in a connection between a pipe and equipment, or between two ducting elements. These four parameters could be encapsulated into a new class called LayeredElementConnectionParameters (similar to the LayeredSetUsageParameters) -- which is used as an optional attribute on this class.
Final Resolution: 1) Current subtypes are by type of connection geometry. This connection geometry information will be moved up to an optional attribute on IfcRelConnectsElements called "ConnectionGeometry ". 2) create a subtype of IfcRelConnectsElements with "IfcRelConnectsLayeredElements" and push these 4 attributes to the subtype.

Action # 1 **Assignee** Liebich **Status** Complete **Resolved in Version** R1.5 - Final

Complete items 1 & 2 described in the resolution.

Issue Number **I - 265** **Issue Date** 9/18/97

Author	See	**Owner**	Liebich	**Status**	Resolved
Schema	IfcProductExt	**Version**	R1.5 - Pre-Final		

Issue Description Class: IfcRelConnectsElements -- What about Peter Muigg's proposal for Logical Connections Enum?

Proposed Solution Incorporate implementers consensis on that -- as discussed in Munich Implementer meeting of 14-Oct-97.

Resolution Reduce the number of options in the Enum (see notes from the 14-Oct-97 meeting).

Study this for a longer term solution in IFC R2.0.

Action # 1 **Assignee** Liebich **Status** Complete **Resolved in Version** R1.5 - Final

Incorporate final agreed enum on IfcRelConnectsElements

Action # 2 **Assignee** Liebich **Status** Incomplete **Resolved in Version** R2.0 - Alpha

Work with implementers to develop a better solution for the long term. See email from R.Steinmann for disucssion on situations current solution will not solve.

Action # 3 **Assignee** See **Status** Incomplete **Resolved in Version** R2.0 - Alpha

Work with implementers to develop a better solution for the long term. See email from R.Steinmann for disucssion on situations current solution will not solve.

Issue Number **I - 266** **Issue Date** 9/18/97

Author	See	**Owner**	Wix	**Status**	Resolved
Schema	IfcProcessExt	**Version**	R1.5 - Pre-Final		

Issue Description Class: IfcWorkTask -- WorkMethod [STRING], TaskCost [IfcCost] -- these are two new attributes (at this late date!).

Proposed Solution Leave them out if not essential.

Resolution These are needed for the concept of ResourseUse -- see other issue on ResourceUse.

Rejected

Issue Number	**I - 267**			**Issue Date**	9/18/97
Author	See	**Owner**	Wix	**Status**	Resolved
Schema	IfcProcessExt	**Version**	R1.5 - Pre-Final		

Issue Description Class: IfcWorkTask -- TaskNumberID [STRING] -- confusing name.

Proposed Solution Rename to WorkTaskID. Note: this follows the naming convention used elsewhere.

Resolution Agreed

Action # 1 **Assignee** Wix **Status** Complete **Resolved in Version** R1.5 - Final

make change as proposed

Issue Number	**I - 268**			**Issue Date**	9/18/97
Author	Liebich	**Owner**	Wix	**Status**	Resolved
Schema	IfcProcessExt	**Version**	R1.5 - Pre-Final		

Issue Description Class IfcRelGroupsWorkTask -- The objectified relationship subtype does not define further information

Proposed Solution Delete and use IfcRelGroups instead; set the "GroupPurpose" to GroupsWorkTasks". Update documentation to make the usage clear.

Resolution [RS] agreed with same reservations about clearly communicating meaning of generalized relationships in specialzed cases where the specialized semantics will now be lost or obscure.

Action # 1 **Assignee** Wix **Status** Complete **Resolved in Version** R1.5 - Final

change as proposed

Issue Number	**I - 269**			**Issue Date**	9/18/97
Author	Liebich	**Owner**	Wix	**Status**	Resolved
Schema	IfcProcessExt	**Version**	R1.5 - Pre-Final		

Issue Description Class IfcWorkTaskSchedule -- has independent ProjectId, but is contained in IfcWorkTask

Proposed Solution Delete ProjectId

Resolution [RS] agreed

Action # 1 **Assignee** Wix **Status** Complete **Resolved in Version** R1.5 - Final

remove attribute as proposed

Issue Number	**I - 270**			**Issue Date**	9/18/97
Author	Liebich	**Owner**	See	**Status**	Resolved
Schema	IfcModelingAidExt	**Version**	R1.5 - Pre-Final		

Issue Description Class IfcLocalPlacement -- error found: the IfcLocalPlacement has to be defined in the IfcKernel, since it is directly reference by another class in IfcKernel -- now the IFC Architecture is violated (see also I-242).

Proposed Solution Bring it back into IfcKernel

Resolution [RS] agreed.

Action # 1 **Assignee** See **Status** Complete **Resolved in Version** R1.5 - Final

eliminate Local placement and reference it from the kernel

Issue Number	**I - 271**			**Issue Date**	9/18/97

Author	See	**Owner**	See	**Status**	Resolved
Schema	IfcModelingAidExt	**Version**	R1.5 - Pre-Final		

Issue Description Class: IfcGridIntersection -- This should be subtyped from IfcReferencePoint so that constrainded placements will really work with Grid intersections (since that placement references ReferencePoints and not ModelingAid).

Proposed Solution Subtype IfcGridIntersection from IfcReferencePoint.

Resolution [TL] I agree

Action # 1 **Assignee** See **Status** Complete **Resolved in Version** R1.5 - Final

change as proposed

Issue Number I - 272 **Issue Date** 9/18/97

Author	See	**Owner**	See	**Status**	Resolved
Schema	IfcModelingAidExt	**Version**	R1.5 - Pre-Final		

Issue Description Class: IfcGridAxis -- This should be subtyped from IfcReferenceCurve so that constrainded placements will really work with Grid Axes (since that placement references ReferenceCurves and not ModelingAid).

Proposed Solution Subtype IfcGridAxis from IfcReferenceCurve.

Resolution [TL] I agree

Action # 1 **Assignee** See **Status** Complete **Resolved in Version** R1.5 - Final

change as proposed

Issue Number I - 273 **Issue Date** 9/18/97

Author	See	**Owner**	See	**Status**	Resolved
Schema	IfcModelingAidExt	**Version**	R1.5 - Pre-Final		

Issue Description Classes: IfcReferencePoint, IfcReferenceCurve, IfcReferenceSurface -- All of these need local placement or an 'implementers convention' that says they are always placed relative to a standard element (Site or Project for example). On reflection, it seems that taking a convention will not work well. In some projects, there may be multiple Sites -- and Project does not have placement.

Proposed Solution Add a mandatory attribute to each -- "RelativePlacement" of type IfcLocalPlacement.

Resolution Agreed.

Action # 1 **Assignee** See **Status** Complete **Resolved in Version** R1.5 - Final

Add attributes as described.

Issue Number I - 274 **Issue Date** 9/18/97

Author	See	**Owner**	Wix	**Status**	Resolved
Schema	IfcDocumentExt	**Version**	R1.5 - Pre-Final		

Issue Description Class: IfcCostScheduleGroup -- GroupID -- no type specified in the EXG (did not check EXP or documentation).

Proposed Solution Include data type.

Resolution Agreed

Action # 1 **Assignee** Wix **Status** Complete **Resolved in Version** R1.5 - Final

Add data tyep and insure that it is consistent for EXP, EXP, DOC

Issue Number I - 275 **Issue Date** 9/18/97

Author	See	**Owner**	Wix	**Status**	Rejected
Schema	IfcDocumentExt	**Version**	R1.5 - Pre-Final		

Issue Description Class: IfcCostSchedule -- ApprovedBy -- I would still argue that 1) cardinality should be a list [0:?] and 2) the data type should be IfcActor because sometimes, approval is needed from an agency (e.g. an organization). While the person that would be used may indeed be part of an organization, it may not be redily apparent. Where the person is important (for accountability/liability), then the SelectType "IfcPersonAndOrganization" will be used. See I-146.

Proposed Solution Make a list [0:?] of IfcActorSelect. Note name change for this SelectType

Resolution Not the same as generalized approval (something for R2/R3), which will then replace this. For R1.5, Approval in this case indicates the person in the organization who approved the costs.

Reject proposed change -- approval to be expanded in R2/R3.

Action # 1 ***Assignee*** Wix ***Status*** Complete ***Resolved in Version*** R1.5 - Final

expand documentation here to clarify the intent as described above.

Action # 2 ***Assignee*** See ***Status*** Complete ***Resolved in Version*** R2.0 - Alpha

Put development of generalized 'Approval' concept in R2 projects list

---

Issue Number ***I - 276*** ***Issue Date*** 9/18/97

Author Liebich ***Owner*** Wix ***Status*** Resolved

Schema IfcDocumentExt ***Version*** R1.5 - Pre-Final

Issue Description Class IfcRelGroupsCostSchedules -- The objectified relationship subtype does not define further information

Proposed Solution Delete and use IfcRelGroups instead; with the "GroupPurpose" set to "GroupsCostSchedules" - update documentation to make the usage clear

Resolution [RS] agreed

Action # 1 ***Assignee*** Wix ***Status*** Complete ***Resolved in Version*** R1.5 - Final

Remove IfcRelGroupsCostSchedules and document use of IfcRelGroups instead.

Issue Number ***I - 277*** ***Issue Date*** 9/18/97

Author Liebich ***Owner*** Wix ***Status*** Resolved

Schema IfcDocumentExt ***Version*** R1.5 - Pre-Final

Issue Description Type IfcCostScheduleOrGroup -- was only needed for the IfcRelGroupsCostSchedules (see I-280).

Proposed Solution Delete it

Resolution [RS] I agree

Action # 1 ***Assignee*** Wix ***Status*** Complete ***Resolved in Version*** R1.5 - Final

delete it as proposed

Issue Number ***I - 278*** ***Issue Date*** 9/18/97

Author Liebich ***Owner*** Wix ***Status*** Resolved

Schema IfcDocumentExt ***Version*** R1.5 - Pre-Final

Issue Description Class: IfcCostScheduleGroup -- This class is subtyped from IfcGroup, therefore: the grouping of IfcCostScheduleElement shall be handled by the IfcRelGroups objectified relationship -- each IfcGroup has a mandatory relationship to IfcRelGroups.

Proposed Solution Delete Element L[0:?] and use IfcRelGroups and an IfcGroup with the "GroupPurpose" set to "CostScheduleGroup". Clarify in the documentation.

Resolution [RS] agreed, but reinforces general issue regarding use of generalized relationships and the need to find a method for redefinition of semantic meaning in derived classes (especially where the classes are many levels below where the generalized relationships are defined).

Final resolution - to be done as proposed.

Action # 1 **Assignee** Wix **Status** Complete **Resolved in Version** R1.5 - Final

change as proposed.

Issue Number ***I - 279*** **Issue Date** 9/18/97

Author See **Owner** Liebich **Status** Resolved

Schema IfcSharedBldgElements **Version** R1.5 - Pre-Final

Issue Description Class: IfcCovering, IfcFloor, IfcRoofSlab, IfcWall -- Layer Information [IfcMaterialLayerSetUsage] -- this reference to materials should be made in type driven Psets.

Proposed Solution Remove from base BuildingElement definitions and establish a convention for references to Materials, MaterialsLayerSets, etc. in type driven PropertySets. See also I-261.

Resolution Compromise resolution: References to Materials from the classes in the statically defined model will remain, but will be modified to allow coordination with Psets. References to materials in Psets will reference an index in the Materials list associated on the static model class. Specific actions:
1) Some objects have multiple materials,but are not layered -- IfcMaterialsList will be added to the Materials part of the PropertyResource - a list of indexes into the IfcMaterialRegistry (see other issue),
2) IfcMaterialSelect will be modified to include IfcMaterialsList and IfcMaterialLayerSet, but NOT IfcMaterial (so that references from Psets can always be an index into a list of materials).
3) references to materials in a Pset will always be an integer index into the MaterialSelect (which of course references materials in the project MaterialRegistry).

Action # 1 **Assignee** Wix **Status** Incomplete **Resolved in Version** R1.5 - Final

complete items 1 & 2

(RS) 26-Nov-97: In Final-Candidate HTML reference docs - item 1 complete. Item 2 note complete as the MaterialSelect still includes IfcMaterial -- which means that references as indexes (from Psets) will not work. This must be a select of LISTs only.

Action # 2 **Assignee** See **Status** Complete **Resolved in Version** R1.5 - Final

complete item 3 as described in the final resolution

Issue Number ***I - 280*** **Issue Date** 9/18/97

Author See **Owner** Liebich **Status** Resolved

Schema IfcSharedBldgElements **Version** R1.5 - Pre-Final

Issue Description Class: IfcWall -- Error found - GenricType -- misspelled.

Proposed Solution Fix spelling

Resolution Agreed

Action # 1 **Assignee** Liebich **Status** Complete **Resolved in Version** R1.5 - Final

change as proposed

Issue Number ***I - 281*** **Issue Date** 9/18/97

Author See **Owner** Forester **Status** Resolved

Schema IfcSharedBldgServiceElem **Version** R1.5 - Pre-Final

Issue Description Class: IfcDistributionElement, IfcElectricalAppliance, IfcFixture -- We had a LONG discussion on these classes in Seattle this week. One conclusion was that these classes are at odds with our stated intent to avoid 'categorizing' element in the class hierarchy (e.g. removal of the IfcLayeredElement and IfcProfiledElement that were in IFC R1.0). This group voiced support for this goal because (they said) we will find real world objects that defy any single classification. Example: a watercooler is BOTH an ElectricalAppliance and a (plumbing) Fixture.

Proposed Solution Continue looking for ways to enable the attachment of multiple extensions onto generic elements (like ElectricalAppliance 'stuff' and Fixture 'stuff'). This should also be consistent with the solution introduced to support multi-functionality in elements (element Groups by functionality). An element can belong to any number of such groups or have any number of the extensions proposed here (e.g. Type "ElectricalAppliance" and "Fixture", each of which results in relating one or more PropertySets.

Resolution [RS] Agreed. However, while in R1.5, extentions for such 'typing' are limited to PropertySets, they will most likely include relationships to objects which define behavior in future releases (e.g. behavior of an "ElectricalAppliance" or a "Fixture". We need to be sure that we have an alternative for 'adding in' such behavior which replaces the inheritence currently used.
[JW] Agreed -- this is related to the multi-functionality problem. Including a supertype which is related to form or function will most likely eventually be removed - as it was for AssembledElement, ManufacturedElement and LayeredElement -- in favor of typing -- multi-typing objects (provided in R1.5) is analogous to multiple functionality.
Final Resolution: leave as it is in R1.5, but study muti-typing along with multi-functionality for R2.0 enhancements.

Action # 1 ***Assignee*** Forester ***Status*** Incomplete ***Resolved in Version*** R2.0 - Alpha

Study multi-typing anlong with multi-functioality (see other issues) in order to propose improvements which truly resolve this issue in R2.0/R3.0.

Action # 2 ***Assignee*** See ***Status*** Incomplete ***Resolved in Version*** R2.0 - Alpha

Study multi-typing anlong with multi-functioality (see other issues) in order to propose improvements which truly resolve this issue in R2.0/R3.0.

Action # 3 ***Assignee*** Liebich ***Status*** Incomplete ***Resolved in Version*** R2.0 - Alpha

Study multi-typing anlong with multi-functioality (see other issues) in order to propose improvements which truly resolve this issue in R2.0/R3.0.

Action # 4 ***Assignee*** See ***Status*** Complete ***Resolved in Version*** R2.0 - Alpha

Add this to the list of projects for R2.0.

Issue Number* I - 282** ***Issue Date 9/18/97

Author See ***Owner*** Forester ***Status*** Resolved

Schema IfcSharedBldgServiceElem ***Version*** R1.5 - Pre-Final

Issue Description Class: IfcMatter -- The BS guys in Seattle had a real problem with this class.

Proposed Solution Use the standard fuel sources instead.

[TL] I agree with recommendation to delete IfcMatter

Resolution Agreed, resolve using 'standard fuel sources' and MeasureWithUnits.

Action # 1 ***Assignee*** Forester ***Status*** Complete ***Resolved in Version*** R1.5 - Final

modify as proposed.

Issue Number* I - 283** ***Issue Date 9/18/97

Author See ***Owner*** Forester ***Status*** Resolved

Schema IfcSharedBldgServiceElem ***Version*** R1.5 - Pre-Final

Issue Description Class: IfcEngineeringMaintenance -- 1) this class definition is DEFINITELY NOT a subtype of IfcControl (as we have defined it) because is does not control, dictate or determine anything in the project. 2) it defines extension information for equipment (note the access space attributes). It should be modeled as a type driven OccurencePropertySet for Equipment and other elements that require maintenance. 3) It should probably also include some information about a maintenance contract and periodic maintenance schedule.

Proposed Solution This is essentially information about the maintenance contacts and access space.

Alt 1) Remodel in the dynamic part of the model as an OccurencePropertySet. Example: see the solution for Door and Window type driven PropertySets which reference an OccurencePropertySet for ManufactureInfo.

Alt 2) See the alternative proposed by email xx-Sep-97 to create a new subtype of IfcObject called "IfcAspect". Maintenance information can be described as a view or "aspect" of an element. Having said that, the Properties associated for this view or aspect could/should use the standard mechanism for associating such 'type driven' propoerties --> back to the first solution alternative proposed.

Resolution The IfcEngineeringMaintenance class really defines maintenance related properties for a piece of equipment (note the access space properties). This will be replaced by an Occurrence Pset reference (from Pset_EquipmentType called Pset_ElementMaintenance (note "Element" rather than "Equipment" so that it can also be used for other subtypes of BuildingElement. This moves these properties from the static part of the model to the dynamic part of the model and can be referenced by any subtype of BuildingElement. Note that Pset_ElementMaintenance should be defined in the IfcProductExt schema so that it can be shared by any building element.

Action # 1 **Assignee** Forester **Status** Complete **Resolved in Version** R1.5 - Final

Define Pset for inclusion in the IfcProductExtension schema as resolved.

Action # 2 **Assignee** See **Status** Complete **Resolved in Version** R1.5 - Final

Insure reference from TypeDriven Psets for elements in Core, Arch and FM models which need maintenance to Pset_ElementMaintenance.

Action # 3 **Assignee** Forester **Status** Complete **Resolved in Version** R1.5 - Final

Insure reference from HVAC Type driven Psets (Equipment, etc.) which need maintenance to Pset_ElementMaintenance.

Action # 4 **Assignee** See **Status** Complete **Resolved in Version** R1.5 - Final

Insure that this Pset is included in the spreadsheet for the IfcProductExtension schema

---

Issue Number I - 284 **Issue Date** 9/18/97

Author See **Owner** Forester **Status** Rejected

Schema IfcSharedBldgServiceElem **Version** R1.5 - Pre-Final

Issue Description Class: IFcEquipment -- TagIdentifier [STRING] -- Name seems redundant. Also, we have a user descriptor on the OwnerIdentification object. So this may be redundant with that.

Proposed Solution 1) rename to EquipmentDescriptor, 2) remove if this the same as the "UserDescriptor" in the IfcOwnerIdentification object.

Resolution Rejected. The "Tag" is different than the UserDescriptor, which is also different than the User Descriptor in the OwnerIdentification.

Issue Number I - 285 **Issue Date** 9/18/97

Author See **Owner** Forester **Status** Resolved

Schema IfcSharedBldgServiceElem **Version** R1.5 - Pre-Final

Issue Description Class: IfcManufacturedElement -- 1) this class definition is DEFINITELY NOT a subtype of IfcControl (as we have defined it). 2) instead, it defines extension information for any manufactured element. It should be modeled as a type driven OccurencePropertySet for Equipment and other elements that are manufactured.

Proposed Solution [RS] note that attaching IfcManufacturedElement at this level of the model (attribute on IfcEquipment) is essentially a workaround for the lack of support for multiple inheritence. This is evident in our difficulty with where to 'place' this class in the model -- it CERTAINLY is NOT a control (it is info about the manufacture - a set of semantically related properties which are related to type).
Alt 1) Remodel in the dynamic part of the model as an OccurencePropertySet referenced from Type driven SharedPropertySets. Example: see the solution for Door and Window type driven PropertySets which reference an OccurencePropertySet for ManufactureInfo.

Alt 2) This is essentially information about the manufacturer. It is not really a control. See the alternative proposed by email xx-Sep-97 to create a new subtype of IfcObject called "IfcAspect". Maintenance information can be described as a view or "aspect" of an element. Having said that, the Properties associated for this view or aspect could/should use the standard mechanism for associating such 'type driven' propoerties --> back to the first solution alternative proposed.

Resolution These properties should be attached through a nested Pset reference from the primary type driven Pset for any element that is manufactured (effectively enabling multiple inheritence). From the Pset_EquipmenType. Include a referenence to an OccurrencePropertySet called Pset_ManufactureInformation as is done with Door and Window types.
The IfcManufacturedElement class really defines information related to the manufacture of an element. This will be re-modeled as an Occurrence Pset referenced from Pset_EquipmentType (and the Shared Psets for other manufactured elements). This Pset will be named

Pset_ManufactureInformation. This moves these properties from the static part of the model to the dynamic part of the model and can be referenced by any manufactured element (generally subtypes of IfcElement). Note: this Pset will be defined in the ProductExt schema so that it can be used by any subtype of IfcElement.

Action # 1 ***Assignee*** Forester ***Status*** Complete ***Resolved in Version*** R1.5 - Final

Define the Pset for inclusion in the IfcProductExtension schema as resolved.

Action # 2 ***Assignee*** See ***Status*** Complete ***Resolved in Version*** R1.5 - Final

Insure reference from TypeDriven Psets for manufactured elements in Core, Arch and FM models to Pset_ElementMaintenance.

Action # 3 ***Assignee*** Forester ***Status*** Complete ***Resolved in Version*** R1.5 - Final

Insure reference from HVAC Type driven Psets (Equipment, etc.) which need maintenance to Pset_ElementMaintenance.

Action # 4 ***Assignee*** See ***Status*** Complete ***Resolved in Version*** R1.5 - Final

Insure that this Pset is included in the Pset spreadsheet for the IfcProductExtension schema

---

Issue Number I - 286 ***Issue Date*** 9/18/97

Author See ***Owner*** See ***Status*** Resolved

Schema IfcArchitecture ***Version*** R1.5 - Pre-Final

Issue Description Class: IfcSpaceProgramme, IfcProgrammeGroup -- During the September domain meetings in Seattle, the group was adamant that we should not use the UK spelling for this class since the UK meaning for this word is different than this use implies (that is, programme means schedule).

Proposed Solution Rename to IfcSpaceProgram and IfcSpaceProgramGroup.

Resolution Agreed

Action # 1 ***Assignee*** See ***Status*** Complete ***Resolved in Version*** R1.5 - Final

change as proposed

Issue Number I - 287 ***Issue Date*** 9/18/97

Author Haiat ***Owner*** See ***Status*** Resolved

Schema IfcModelingAidExt ***Version*** R1.5 - Pre-Final

Issue Description [raised by J.C. Haiat - logged by R.See]
IfcDesignGrid and IfcGridLevel -- It was discussed in the September Implementers meeting that it might be better to reverse the relationships "HasAxes" and "HasLevels" in the Design Grid entities.

Proposed Solution Please reverse them.

Resolution Agreed

Action # 1 ***Assignee*** See ***Status*** Complete ***Resolved in Version*** R1.5 - Final

reverse the direction for these relationships

Issue Number I - 288 ***Issue Date*** 9/18/97

Author Haiat ***Owner*** Liebich ***Status*** Resolved

Schema IfcGeometryResource ***Version*** R1.5 - Pre-Final

Issue Description [raised by J.C. Haiat, entered by R.See]
The current mechanism for defining walls is cumbersome in a number of cases.

Proposed Solution We need to be able to extrude Walls vertically and allow them to be 'trimmed' by floor and Ceiling planes.

In an attempt to generalize the solution, the following compromise is proposed by RS.
1) extrusion along the path will be retained (since it is 'most' appropriate in some cases) (see also 3)
2) a top and bottom clipping "curve" will be defined along with an extrusion direction vector (note

that these curves are aligned with the path). The receiving application must extrude these curves along the matched vectors creating clipping surfaces. The Wall, Floor, Roofslab (or whatever uses this AttDrivenShape type (to be called "IfcAttDrivenTrimmedExtrudedSolid") will then be trimmed, eliminating the portions above the top clipping surface and below the bottom clipping surface.
3) A "Geometry Use" case will be added for Walls -- where the extrusion direction is perpendicular to the wall path (e.g. vertical).

Resolution Final Resolution: 1) Vertical extrusion is an extension that we will consider in R2.0. For R1.5, we will only support extrusion along the path. Note that the advantages cited for vertical extrusion are now supported through the abiliy to trim at the ends of the extrusion (as well as top and bottom).
2) Trimming will be done by a ClippingHalfSpaces = LIST [0:?] IfcHalfSpaceSolid (an IfcHalfSpace is defined by a surface and a BOOLEAN indicating which side of the surface is solid).

Note: this is not limited to top and bottom. This will allow trimming at the ends of walls as well (to allow the mitered corners shown in the implementer's meeting on 9-Sep-97).

Action # 1 **Assignee** Liebich **Status** Complete **Resolved in Version** R1.5 - Final

Created two new subtypes: - IfcAttDrivenClippedExtrudedSolid, IfcAttDrivenClippedRevolvedSolid

each getting the attribute: ClippingHalfSpaces : LIST [1:?] OF IfcHalfSpaceSolid;

Also requires adding an additional Entity: IfcHalfSpaceSolid (BaseSurface : IfcSurface; AgreementFlag : BOOLEAN;)

Action # 2 **Assignee** Liebich **Status** Incomplete **Resolved in Version** R1.5 - Addend

Add a new "Geometry Use" case for vertically extruded wall segments -- investigate the consequences of connecting such elements at the end points of their paths, rather than the endpoints of their extrusions.

Issue Number	**I - 289**			**Issue Date**	9/30/97
Author	See	**Owner**	Liebich	**Status**	Resolved
Schema	IfcProductExt	**Version**	R1.5 - Pre-Final		

Issue Description IfcRelConnectsElements -- The agreed Dependency flags (one each for RelatingObject and RelatedObject) have been left out.

Proposed Solution Add two dependencey flag (BOOLEAN) attributes (RelatingObjectDependent, RelatedObjectDependent) as agreed in email thread from mid-September -- at the location where the "Dependency" flag was in the Pre-Beta.

Resolution Agreed -- note that these flags are on IfcRelationship.

Action # 1 **Assignee** Liebich **Status** Complete **Resolved in Version** R1.5 - Final

change as proposed.

Issue Number	**I - 290**			**Issue Date**	10/15/97
Author	Shulga	**Owner**	Liebich	**Status**	Rejected
Schema	IfcGeometryResource	**Version**	R1.5 - Pre-Final		

Issue Description IfcBoundingBox should be renamed because BoundingBox has special meaning to me for spatial comparisons of min/max points.

Proposed Solution Rename to IfcBlockShapeRep

Resolution Not convinced that this must be done

Issue Number	**I - 291**			**Issue Date**	10/15/97
Author	Shulga	**Owner**	Liebich	**Status**	Resolved
Schema	IfcGeometryResource	**Version**	R1.5 - Pre-Final		

Issue Description IfcAttributeDrivenProfileDef - Arbitrary profile def. Should not have a descriptor based on

products (geometry should be separated from the Semantic model objects).
Also the 'geometry use' definitions need some improvements - see proposed edits in document sent to TL.

Proposed Solution Remove the 'Descriptor' attribute from the model and consider the edits proposed in the doc given to TL.

Resolution Agreed.

Action # 1 **Assignee** Liebich **Status** Complete **Resolved in Version** R1.5 - Final

Make changes as described.

Issue Number	**I - 292**			**Issue Date**	10/15/97
Author	Shulga	**Owner**	Liebich	**Status**	Resolved
Schema	IfcGeometryResource	**Version**	R1.5 - Pre-Final		

Issue Description AttDrivenExtrudedSolid / AttDrivenExtrudedSegment
AttDrivenRevolvedSolid / AttDrivenRevolvedSegment - it is a real problem to have only one placement for the AttDrivenExtrudedSolid -- should have a placement for each segment.

Proposed Solution add a placement for each segement and remove the one for the extruded solid container.

Resolution Agreed, NOTE a WHERE rule will have to be added which insures that the direction of extrusion axes (Z-axis) are equivalent.

Action # 1 **Assignee** Liebich **Status** Complete **Resolved in Version** R1.5 - Final

Make changes as proposed.

Issue Number	**I - 293**			**Issue Date**	10/15/97
Author	Shulga	**Owner**	Liebich	**Status**	Resolved
Schema	IfcGeometryResource	**Version**	R1.5 - Pre-Final		

Issue Description IfcMorphedExtrudedSegment - the descriptions are confusing. Is the intent that the resulting surfaces must be planar?

Proposed Solution Add an informal proposition to clearly state this intention. See wording proposed in doc sent to TL.

Resolution Agreed.

Action # 1 **Assignee** Liebich **Status** Complete **Resolved in Version** R1.5 - Final

Make changes as proposed.

Issue Number	**I - 294**			**Issue Date**	10/15/97
Author	Shulga	**Owner**	Liebich	**Status**	Resolved
Schema	IfcGeometryResource	**Version**	R1.5 - Pre-Final		

Issue Description IfcAttDrivenRevolvedSolid - "Radius" is meaningless here. What you really need is an axis. Additionally, the geometry is defined in the Segements, so the axis is needed there not in the aggregator.

Proposed Solution Remove the "Radius" attribute and reference a placement which defines the revolution axis. Each segment would then need a StartAngle and SweepAngle (second one is inherited from IfcRevolvedAreaSolid).

Resolution Partially agreed. NOTE: We want to insure that the Axis for each segment is the same. NS would like to insure the segments reference a common placement through a WHERE rule in the IfcAttDrivenRevolvedSolid.

Action # 1 **Assignee** Liebich **Status** Complete **Resolved in Version** R1.5 - Final

Make changes as discussed in Munich meeting 15-Oct.

Issue Number	**I - 295**			**Issue Date**	10/15/97
Author	Shulga	**Owner**	Liebich	**Status**	Resolved

Schema IfcGeometryResource **Version** R1.5 - Pre-Final

Issue Description In the IfcAttDrivenProfileDef - these shapes need a distinction between use as a curve (for swept shells - future) and use as areas (for swept solids - now).

Proposed Solution Add back the attribute "ProfileType" [enumeration for IfcProfileTypeEnum (Curve, Area)] on IfcAttDrivenProfileDef.

Resolution Agreed.

Action # 1 **Assignee** Liebich **Status** Complete **Resolved in Version** R1.5 - Final

Make the changes as proposed.

Issue Number I - 296 **Issue Date** 10/15/97

Author See **Owner** Liebich **Status** Resolved

Schema IfcGeometryResource **Version** R1.5 - Pre-Final

Issue Description What happened to the TaperedExtrusion segment we agreed in September -- was in the Pre-Beta and then disappeared in the Pre-Final

Proposed Solution "IfcAttDrivenTaperedExtrudedSegment" needs to be added back in as agreed in discussions after Pewsey.

Resolution Agreed.

Action # 1 **Assignee** Liebich **Status** Complete **Resolved in Version** R1.5 - Final

Add it in as discussed and to be consistent with other extrusion 'segments'.

Issue Number I - 297 **Issue Date** 9/30/97

Author Cole **Owner** Wix **Status** Resolved

Schema IfcProcessExt **Version** R1.5 - Pre-Final

Issue Description I see that IfcWorkGroup is no longer an IfcProcess. Instead it is an IfcGroup. I think this is a problem.

In costing and scheduling, we often want to break down tasks to a finer granularity than we will want to schedule. Therefore, we will want to schedule a grouping of tasks, rather than each elemental task.
This is no longer possible since an IfcWorkGroup does not have
"IfcProcess" capabilities. This will especially make it difficult to share task information between costing and scheduling.

Proposed Solution Make IfcWorkGroup a process.
[RS] Alt1) what if the relationship to IfcWorkTaskSchedule were reversed and made into a List (e.g. SchedulesWorkTasks ::LIST[1:N] IfcWorkTask). Drawback: This does not guarantee 1to1 correspondence between an IfcGroup used in a Cost Schedule and a group schedule by this LIST.
[RS] Alt 2) reverse the relationship and redirect to IfcWorkGroup - meaning that you can only schedule groups of one or more tasks. Note - this does not necessarily mean that the IfcWorkGroup must be a Process.
[RS] Alt 3) reverse relationship and redirect to a Select type "IfcWorkTaskOrGroupSelect"

Resolution Note: For any of the proposed solutions, since the Schedule object would be used for either Tasks or Groups of tasks, the schedule class should be renamed to "IfcWorkSchedule" -- where a group will have one or more tasks.
Final resolution - will use alternative 3 and change the name of the schedule to "IfcWorkSchedule".

Action # 1 **Assignee** Wix **Status** Complete **Resolved in Version** R1.5 - Final

make changes as resolved.

Issue Number I - 298 **Issue Date** 10/15/97

Author See **Owner** Wix **Status** Resolved

Schema IfcPropertyResource **Version** R1.5 - Pre-Final

Issue Description There are a VERY LARGE number of material references in the PropertySets which are now left to uncoordinated STRING values. This will lead to chaos in trying to coordinate material designations between applications.

Proposed Solution Create a project materials registry and allow indexed use of material definition entries (as with Project teams members and applications) from within PropertySets.

Resolution See also I-261 and I-304

Agreed.

1) Add IfcProjectMaterialsRegistry to the PropertyResource (referenced by IfcProject).
2) Add IfcMaterialList to the PropertiesResource (to be referenced by any object having none layered materials. Each entry in a MaterialLayerSet or a MaterialList will be an integer index into the Registry described in 1.
3) Update all material references in PropertySets to use references into the MaterialLayerSet or MaterialSet related to the base object (see reference on IfcBuildingElement). These references will be of type INTEGER (an index into the list of materials for this object).

Action # 1 **Assignee** Wix **Status** Complete **Resolved in Version** R1.5 - Final

1) Add IfcProjectMaterialsRegistry to the PropertyResource.

Action # 2 **Assignee** Wix **Status** Complete **Resolved in Version** R1.5 - Final

2) Add IfcMaterialList to the PropertiesResource (to be referenced by any object having none layered materials. Each entry in a MaterialLayerSet or a MaterialList will be an integer index into the Registry described in 1.
Done originally at IfcMaterialComposite -- then name changed to IfcMaterialList as result of I-315.

Action # 3 **Assignee** See **Status** Complete **Resolved in Version** R1.5 - Final

3) Update all material references in PropertySets to use references into the MaterialLayerSet or MaterialSet related to the base object (see reference on IfcBuildingElement). These references will be of type INTEGER (an index into the list of materials for this object).

Issue Number I - 299 **Issue Date** 10/15/97

Author See **Owner** Liebich **Status** Resolved

Schema IfcPropertyTypeResource **Version** R1.5 - Pre-Final

Issue Description We haven't captured the "TypeDefName" and therefore could not even query the name of a TypeDef

Proposed Solution Add an attribute "TypeDefName" and a query in the default interface.

Resolution Agreed

Action # 1 **Assignee** Liebich **Status** Complete **Resolved in Version** R1.5 - Final

change as proposed

Issue Number I - 300 **Issue Date** 10/15/97

Author See **Owner** Liebich **Status** Rejected

Schema IfcPropertyTypeResource **Version** R1.5 - Pre-Final

Issue Description In developing the Type Definitions and associated PropertySets, it became apparent that the 'Parent' PropertySet should be listed with any other nested PropertySets. Otherwise, it is too difficult to tell if an attribute is already covered -- the relationship that is included in the TypeDef object makes the aggregate collection of properties too separate.

Proposed Solution Eliminate the relationship to parent in favor of including the parent as a referenced PropertySet (see examples for Walls, Doors, Windows sent to implementers on 19-Oct).

Resolution NOTE: this is already covered in the descriptions and resolutions to I-306. Referenced from there, but Rejected here as it is already covered.

Issue Number I - 301 **Issue Date** 10/15/97

Author See **Owner** Liebich **Status** Resolved

Schema IfcPropertyTypeResource **Version** R1.5 - Pre-Final

Issue Description IfcPropertyDef - we need to be able to include LISTs and SETs of properties within a PropertySet.

This issue re-opened in telecon 26-Nov - using reference Psets for this is VERY HARD TO FOLLOW. Furthermore, we need to include variable length LISTs/SETs of same data types. How do we specify this in our spreadsheets where we have to pre-declare everything ??

Proposed Solution Add two subtypes to IfcPropertyDef - aggregators - one for SETs and one for LISTs

Resolution Aternative solution agreed. Instead, we will use the ability to nest Psets - explained in the documentation.
For example, a candiate List Property in a Pset will be defined as data type [[LIST [x:y] OF IfcProperty]]. The Model Guide documentation will explain to implementers that this should be implemented as a nested PropertySet -- either Shared or Occurrence depending on whether the data is shared by all occurrences or varies with each. NOTE: this solution will be used for each of LIST, SET, BAG, ENUM

This issue re-opened in telecon 26-Nov - using reference Psets for this is VERY HARD TO FOLLOW.

Action # 1 **Assignee** See **Status** Complete **Resolved in Version** R1.5 - Final

Update all Core model and Architecture related PropertySets which currently include LIST, SET, BAG or Enum

Action # 2 **Assignee** Forester **Status** Complete **Resolved in Version** R1.5 - Final

Update all Building Service related PropertySets which currently include LIST, SET, BAG or Enum

Action # 3 **Assignee** Yu **Status** Complete **Resolved in Version** R1.5 - Final

Update all FM related PropertySets which currently include LIST, SET, BAG or Enum

Action # 4 **Assignee** See **Status** Incomplete **Resolved in Version** R1.5 - Final

Include in the Model Guide -- the interpretation instructions to implementers as described in Resolution.

Issue Number I - 302 **Issue Date** 10/15/97

Author See **Owner** Liebich **Status** Resolved

Schema IfcPropertyTypeResource **Version** R1.5 - Pre-Final

Issue Description We need a way to reference some types of geometry from within PropertySets. For example, the need to include a Polyloop profile as in the PropertySets for Doors and Windows.

Proposed Solution Alt 1) Create a subtype of IfcPropertyDef which wraps selected Geometry entities -- for example the PolyLoop used in the Door and Window PropertySets -- called "IfcProfileProperty".
Alt 2) subtype IfcGeometryRepresentationItem from IfcPropertyDef -- in which case we could include any type of geometry in a PropertySet
Alt 3) require that any use of geometry in PropertySets be defined within an IfcProductShape, which is already subtyped from IfcPropertyDef. This was considered in the examples listed, but considered to be too heavy for including a simple Polyloop.

Resolution Will add a ProjectUniqueID to the two types of ComponentShape (which is a select type) --> IfcShapeResult and IfcShapeBody -- in the ProductShape part of IfcPropertyType schema -- so that these can be referenced from PropertySets -- using the IfcObjectReference subtype of IfcProperty. This means that the "Frame" of a window can point directly to the geometry shape component used for representation.

Action # 1 **Assignee** Liebich **Status** Complete **Resolved in Version** R1.5 - Final

change as resolved

Action # 2 **Assignee** See **Status** Complete **Resolved in Version** R1.5 - Final

Update all PropertySets to use the new type added in action 1

Issue Number I - 303 **Issue Date** 10/15/97

Author See **Owner** Liebich **Status** Resolved

Schema IfcSharedBldgElements **Version** R1.5 - Pre-Final

Issue Description The "Geometry Use" sections of the documentation for IfcDoor and IfcWindow have not been completed. It is IMPERATIVE that we include these sections in order to eliminate ambiguity regarding the 'standard way' to use geometry for the IfcProductShape of these and other entities.

Proposed Solution Develop these sections of documentation before the Final Specifications are published.

Resolution Agreed.

(RS) 26-Nov-97: Still needed for Door, Window, BuildingStorey, Building, Site. Should probably should improve for Beam (horzontal extrusion - given definition).

Action # 1 ***Assignee*** Liebich ***Status*** Incomplete ***Resolved in Version*** R1.5 - Final

Develop additional documentation as described.

(RS) 26-Nov-97: Not done in Final-Candidate HTML reference docs.

Issue Number ***I - 304*** ***Issue Date*** 10/15/97

Author	See	***Owner***	Wix	***Status***	Rejected
Schema	IfcPropertyResource	***Version***	R1.5 - Pre-Final		

Issue Description We MUST, MUST, MUST define a registry of materials for the project (as with the IfcProjectAppRegistry and IfcTeamRegistry). The number of material references that are currently of type STRING in the PropertySets demands it. NOTE: it is not necessary to reference them using integers as with the examples. It IS necessary that the list of Materials is non-redundant and that any material can be referenced from a PropertySet.

Proposed Solution Insure a SIMPLE method to develop a registry of unique material designations that can be referenced from PropertySets

Resolution NOTE: this is essentially already covered in the resolution to I-298. Rejected here as it is already covered.

Issue Number ***I - 305*** ***Issue Date*** 10/15/97

Author	See	***Owner***	Liebich	***Status***	Resolved
Schema	IfcProductExt	***Version***	R1.5 - Pre-Final		

Issue Description We have not incorporated an enum for connections between path based elements (extruded) into the IfcRelConnectsElements class. However instances of this class will be used to connect non path based elements also (e.g. connecting two pieces of Equipment (equipment is not path based). The enum inappropriate for such connections.

Proposed Solution Subtype IfcRelConnectsPathElements from IfcRelConnectsElements, which will include the enum. The "LayeredElementConnectionParameters::IfcLayeredElementConnectionParameters" (see resolution to I-264) should also be moved down to this subtype since LayeredElements will always be path based.

Resolution Agreed

Action # 1 ***Assignee*** Liebich ***Status*** Complete ***Resolved in Version*** R1.5 - Final

Create the subtype and move the two attributes down

Issue Number ***I - 306*** ***Issue Date*** 10/15/97

Author	See	***Owner***	See	***Status***	Resolved
Schema	IfcPropertyTypeResource	***Version***	R1.5 - Pre-Final		

Issue Description We have two dilemas to resolve in the definition of type driven PropertySets and relationships between these (using the nesting references and the "Parent" references).
1) excluding references to Parent PropertySets (Pset) from a Pset definition makes it too obscure. It is VERY difficult to 'see' when some obvious properties are missing from a Pset -- that they are included in a Parent Pset UNLESS the reference to the Parent Pset is included as a nested reference.
2) Nested references to type driven OccurrencePropertySets from SharedPropertySets will have to be of type STRING, since there will be a different one for each occurrence of the type.

Proposed Solution 1) since a nested reference (within a Pset) is funtionally equivalent to the Parent reference (defined overtly in the TypeDef), we should eliminate the second in favor of the first to enhance

common understanding of the models.
2) change all nested references to type driven OccurrencePropertySets from SharedPropertySets to IfcString. This STRING will contain the name of the Pset, which is in the list of Occurrence Psets attached to the "typed" object. Applications will need to search this OccurrencePropertySet list (at the IfcProduct level) to find the named Pset.
NOTES: 1) this underscores the importance of including the Pset name in the Pset definition. 2) We cannot use IfcOccurrencePropertySet or IfcObjectReference here because it is a "1 to N" relationship between the referencing Pset and the occurrence values for multiple instances.

Resolution see also I-300
1) Agreed - reference from TypeDef changed to "GenericTypeRef" (not parent) as this was included so that TypeDefs for Specific types could reference their GenericType.
2) Agreed -- documentation should make this clear with diagrams. Note that an application interpreting an object with such a Pset (containing a reference to an occurrence Pset) will have to search the Occurrence Pset list (at the IfcObject level) of the 'typed' object -- to find the one for which the "Descriptor" (should be "PsetName") matches the STRING value in the reference.

Action # 1 ***Assignee*** Liebich ***Status*** Complete ***Resolved in Version*** R1.5 - Final

Change the Parent Pset reference in the TypeDef class to an optional reference to the Generic Type definition associated with this type -- NOTE: only used in the case of Specific Type Defs.

Action # 2 ***Assignee*** See ***Status*** Complete ***Resolved in Version*** R1.5 - Addend

Change all references to Occurrence Psets (in Psets) to be of type IfcString.

Action # 3 ***Assignee*** See ***Status*** Incomplete ***Resolved in Version*** R1.5 - Addend

Enhance Model Guide documentation regarding different types of nested references from with Psets -- using diagrams and clarifying differences between references to Shared Psets and references to Occurrence Psets.

Issue Number I - 307 ***Issue Date*** 10/15/97

Author See ***Owner*** Wix ***Status*** Resolved

Schema IfcPropertyResource ***Version*** R1.5 - Pre-Final

Issue Description Class: IfcMaterial -- "SystemClassification::IfcMaterialClassification"
1) this attribute name is misleading
2) it is probably better to allow ofr multiple classifications here as in classification of objects. Material classifications will be different for different regions of the world.

Proposed Solution 1) Change name of attribute to "MaterialClassification"
2) Can we simply add "Classification" instead? Only if we modify the IfcClassification class to allow for multipart "Notation"s (currently only a single string) -- in this case, we need to use a "MainCategory"

Resolution 1) Agreed - change it.
2a) modify Notation to breakdown into 3 fields (as in the ISO simple classification scheme). Field 2 and 3 should be optional
2b) change the attribute on IfcMaterial (and its data type) to "MaterialClassification::IfcMaterialClassificationList"
(JW-980510) Move the classification relation to from IfcMaterialList to fcMaterial. This enables the IfcMaterialList to be deleted and makes material classification work in the same way as other classification forms. Subtyping from IfcProperty should also be extended to all of the main entities in the Material model.

Action # 1 ***Assignee*** Wix ***Status*** Incomplete ***Resolved in Version*** R1.5 - Final

modify the Material and Classification sheets of this schema as resolved.

Issue Number I - 308 ***Issue Date*** 10/20/97

Author See ***Owner*** See ***Status*** Deferred to R2.0

Schema IfcPropertyResource ***Version*** R1.5 - Pre-Final

Issue Description We have lost the ability to "TYPE" properties -- examples where this was done in R1.0 = IfcActor (now a select type --> IfcPerson, IfcOrganization).
Other examples where this is desirable = IfcCost, IfcMaterial

Proposed Solution Either associate TypeDef and OccurrencePsets at these properties specifically, or with

IfcProperty (their supertype).

Resolution This would require enabling TypeDefinitions for IfcProperty (and subtypes) -- which seems a bit premature for R1.5. Therefore, we will defer to R2.0.

Issue Number	***I - 309***			***Issue Date***	10/15/97
Author	See	***Owner***	See	***Status***	Resolved
Schema	IfcPropertyTypeResource	***Version***	R1.5 - Pre-Final		

Issue Description We have several examples where we need to include enumerations as the data type for properties in Psets.

Proposed Solution Alt 1) Comma delimited values, stored in a STRING, prefaced with a selection for this occurrence (of the Pset). Agreed values to be published in the IFC Specifications will allow conformance testing.
Example: (2, value1, value2, value3)
Alt 2) define the range of values in a Pset and then refernce a value from the subject Pset. Note: this means that the subject Pset will need 2 values for each enum, one referencing the Pset_XxxEnum and the other with the selected value index (index into the list of values in the Pset_XxxEnum.

Resolution Will go for alternative 2.

Note: this solution superseeded by agreement between RS and JF. Enums will be documented in the same way as LIST, SET and BAG in Psets (see resolution to I-301) --> they will be defined with the list of valid values in the data type declaration. Implementers will be instructed in the Model Guide documentation to implement each of these types as nested Psets.

Action # 1 ***Assignee*** See ***Status*** Eliminated ***Resolved in Version*** R1.5 - Final

Define Pset_XxxxEnum for each of the enums currently defined in Core and Arch Psets.

For each, define the range of values in a Pset and then refernce a value from the subject Pset. Note: this means that the subject Pset will need 2 values for each enum, one referencing the Pset_XxxEnum and the other with the selected value index (index into the list of values in the Pset_XxxEnum.

Action # 2 ***Assignee*** Forester ***Status*** Eliminated ***Resolved in Version*** R1.5 - Final

Define Pset_XxxxEnum for each of the enums currently defined in HVAC Psets.

For each, define the range of values in a Pset and then refernce a value from the subject Pset. Note: this means that the subject Pset will need 2 values for each enum, one referencing the Pset_XxxEnum and the other with the selected value index (index into the list of values in the Pset_XxxEnum.

Action # 3 ***Assignee*** Yu ***Status*** Eliminated ***Resolved in Version*** R1.5 - Final

Define Pset_XxxxEnum for each of the enums currently defined in Core and Arch Psets.

For each, define the range of values in a Pset and then refernce a value from the subject Pset. Note: this means that the subject Pset will need 2 values for each enum, one referencing the Pset_XxxEnum and the other with the selected value index (index into the list of values in the Pset_XxxEnum.

Issue Number	***I - 310***			***Issue Date***	10/28/97
Author	Child	***Owner***	See	***Status***	Deferred to R2.0
Schema	All Schemata	***Version***	R1.5 - Pre-Final		

Issue Description Subtyping of Objectified relationship in order to further specialize the RelatingObject/RelatedObjects violates the "Liskov substitution" tenant in object oriented software design --> that is: the interface contract set by the supertype is broken by further specialization in the subtypes.
See email discussion thread beginning 28-Oct-97 entitled "Modelling of relationships in IFCs"

Proposed Solution Eliminate this subtyping and limit the object types in the desired circumstances through the use of WHERE rules.

Resolution Eliminating this from the model now would take months. We must find a workaround and look at evolving the model to eliminate this (apparent) design shortcoming. 1) TL will contact Martin at Nemetschek to find out how he resolved this problem in his programming and will look at adding to our documentation -- implementer advice about how to deal with it. T.Child should be review group lead for proposed implementer advice (and invited to contribute to it). 2) RS to add to list of R2.0 projects, search for longer term solution.

Resolution for R2.0 --> remove the relationships RelatingObject and Related Object(s) in the abstract supertypes --> IfcRelationship1to1 and IfcRelationship1toN. This will eliminate the redeclaration of these relationships in the subtype. NOTE: will add to the modeling guidelines that subtyped Objectified Relationships must not redeclare the RelatingObject and Related Object(s).

Action # 1 ***Assignee*** Liebich ***Status*** Eliminated ***Resolved in Version*** R1.5 - Final

Work out implementer advice (with help from Martin and T.Child) .

Action # 2 ***Assignee*** See ***Status*** Complete ***Resolved in Version*** R2.0 - Alpha

Add to list of R2.0 projects --> research and development of longer term solution (R2.0 and beyond)

Action # 3 ***Assignee*** Liebich ***Status*** Incomplete ***Resolved in Version*** R2.0 - Alpha

Make the changes to the Kernel schema as described above in the resolution for R2.0.

Issue Number ***I - 311*** ***Issue Date*** 11/28/97

Author Shulga ***Owner*** See ***Status*** Resolved

Schema IfcGeometryResource ***Version*** R1.5 - Pre-Final

Issue Description Class: IfcAxis2Placement3D - defaulting only one of axis or ref_direction can lead to invalid transform matricies.

Proposed Solution In IfcAxis2Placement3D: either both axis and ref_direction should be defaulted, or none. A rule should be added to that effect. That should replace the 'adjusted as needed' phrase.

Resolution Agreed. Will add a where rule which requires both values or neither value.

Action # 1 ***Assignee*** Liebich ***Status*** Complete ***Resolved in Version*** R1.5 - Final

Add 'where rule' (WR) and note that this is different than STEP P42.

Issue Number ***I - 312*** ***Issue Date*** 10/29/97

Author Muigg ***Owner*** Liebich ***Status*** Resolved

Schema IfcKernel ***Version*** R1.5 - Pre-Final

Issue Description LocalPlacement is mandatory for all Products, also for Site. The PlacementRelTo attribute is also mandatory at IfcLocalPlacement. Therefore a Site MUST be placed relative to another Product or ModelingAid.

Proposed Solution Make PlacementRelTo at IfcLocalPlacement optional. Indication: if set, placement is relative, if not set, placement is absolute (WCS).

[RS email - 1-Nov] This recommendation sounds good initially, but there is a catch = we don't have a WCS established for the project. This is because the Project has no placement. It also points out another 'gotcha' in our model that would have come up at some point = a project may have multiple Sites, each of which has a different reference geographic reference point. To remedy this and enable your recommended solution I suggest the following changes in the R1.5 Final Models (NOT FOR ACS):

IfcProject:
1) Add the attributes IfcReferenceLongitude, IfcReferenceLatitude and IfcReferenceElevation (currently defined for IfcSite)
2) Add the attribute ProjectWCS of type IfcAxis2Placement3D. This placement will be relative to the geographic reference point established by the attributes above and will establish the WCS for the project.

IfcObjectsWithPlacement
3) Add IfcProject to this select type (so that objects can be placed relative to the project WCS)

IfcSite
4) Remove the attributes IfcReferenceLongitude, IfcReferenceLatitude and IfcReferenceElevation (now moved to the Project)

5) Add the attribute LocalPlacement of type IfcLocalPlacement (by convention, Sites will be placed relative to the Project WCS).

I think that this will cover it and also believe that this is a better solution all around. Now placement of sites is just as with any other product and the Project object is the only special case. Additionally, Modeling Aids (like the DesignGrid) can be placed relative to the Project WCS.

This brings up a very good point! This means that the RelativeTo attribute of LocalPlacement could remain mandatory. This is because the only special case (IfcProject) does not use LocalPlacement, but uses the IfcAxis2Placement3D directly. Two sides to this: a) making the attribute optional (and taking the convention that this means placement relative to the project WCS) is simpler, b) it is also slightly more ambiguous.

Resolution 1)WCS must be established on the IfcProject level. Sites must therefore be placed relative to the Project. We consider latitude/longitude/elevation - Geographic reference point to be inadequate for GIS placement. Therefore we will leave the Geographic reference point on the site as approximate and not reconciled to the exact placement -- for use by applications related to sun angle,climate, etc. We will wait to add GIS palcement on IfcProject in R2.0. 2) IfcProject will be added to IfcObjectsWithPlacementSelect, 3) placement for site will use the normal LocalPlacement w/ WR that will force placement relative to Project. 4) PlacementRelTo on LocalPlacement will be made optional -- with the convention that, where not included, placement is in the WCS (as established by the IfcProject LCS).

Action #	1	**Assignee**	Liebich	**Status**	Complete	**Resolved in Version**	R1.5 - Final

make changes in IfcKernel (items 2,3,4)

Action #	2	**Assignee**	See	**Status**	Complete	**Resolved in Version**	R2.0 - Alpha

Add to R2.0 list of projects -- addition of GIS placement on IfcProject

---

Issue Number	**I - 313**			**Issue Date**	11/25/97
Author	See	**Owner**	See	**Status**	Resolved
Schema	IfcKernel	**Version**	R1.5 - Pre-Final		

Issue Description There is no inverse relationship from IfcProject to IfcRelContains. This means that the only way to find out all the elements 'contained' in a project (say Building), is to iterate over the IfcRelContains rels and find the ones which reference the Building as the RelatingObject. There is no way to query a project for all the objects it contains.

This is not a problem for IfcBuilding, IFcBuildingStorey or IfcSpace as the inverse relationship has been declared for each of these.

Proposed Solution The inverse relationships we had in the PreBeta(Contains and ReferencedBy) should be replaced -- inverse for the IfcRelContains relationships rather than the relationships directly to other objects (as before).

Resolution Agreed.

Action #	1	**Assignee**	Liebich	**Status**	Incomplete	**Resolved in Version**	

Make the changes as agreed

Issue Number	**I - 314**			**Issue Date**	11/25/97
Author	See	**Owner**	See	**Status**	Resolved
Schema	IfcMeasureResource	**Version**	R1.5 - Pre-Final		

Issue Description As discussed in Frankfurt meetings -- we NEED a measure value that we can use in Psets for INTEGER. Currently there is no way to do an INTEGER in Psets (only REAL and NUMBER).

Proposed Solution Add a Measure Value called IfcIntegerCountMeasure of type INTEGER.

Resolution Agreed.

Superseded by more comprehensive solution in I-316.

Action #	1	**Assignee**	Liebich	**Status**	Eliminated	**Resolved in Version**	R1.5 - Final

Add entity as defined.

Issue Number ***I - 315*** ***Issue Date*** 11/25/97

Author See ***Owner*** See ***Status*** Resolved

Schema IfcPropertyResource ***Version*** R1.5 - Pre-Final

Issue Description IfcCompositeMaterial -- this is a BAD name for the intended purpose this class (see description in I-261). This is supposed to be a simple LIST of materials = IfcMaterialList. The work "Composite" in the US has a specific meaning -- as in fused or structurally combined materials -- as are used in high end manufacturing. This is NOT what we mean when we want to include a list of materials for a Door or Window (where one material is the frame, another is the glazing, another is the panel, etc.).

Proposed Solution Change the name to IfcMaterialList

Resolution Agreed.

Action # 1 ***Assignee*** Wix ***Status*** Complete ***Resolved in Version*** R1.5 - Final

change as proposed

Issue Number ***I - 316*** ***Issue Date*** 9/18/97

Author All STF ***Owner*** See ***Status*** Resolved

Schema IfcMeasureResource ***Version*** R1.5 - Pre-Final

Issue Description Currently, the following data types are EXCEEDINGLY difficult to represent in PropertySets: STRING, INTEGER, BOOLEAN. Additionaly, it would be good if we had a simple REAL that we could use in Psets.

Note: There is currently no data types in the Measure schema (all simple properties are of type IfcMeasureValue) for STRING, INTEGER, BOOLEAN.

Proposed Solution Add base data types for these in either the Measure or Utility Resources

Resolution Agreed - see also I-314 for specific issue regarding INTEGER.

Add these 4 data types (IfcString, IfcInteger, IfcReal, IfcBoolean) to the IfcMeasureResource (must be subtyped from IfcMeasureValue" since this is the data type for IfcSimpleProperties to be included in Psets).

Action # 1 ***Assignee*** Liebich ***Status*** Complete ***Resolved in Version*** R1.5 - Final

Make additions as described.

Issue Number ***I - 317*** ***Issue Date*** 11/26/97

Author See ***Owner*** See ***Status*** Resolved

Schema IfcSharedBldgElements ***Version*** R1.5 - Final Candi

Issue Description Component lists for Doors and Windows are not correct. Appear to be based on the ACS demos subset rather than the R1.5 definitions.

Proposed Solution For Doors: Lining, Frames, Panels, Trim, Hardware [[Component breakdown:
Door < Lining + (Panels < Panels + Openings + OpeningFiller) + Trim + Hardware]]

For Windows: Lining, Panels, Frames, Glazing, Trim, Hardware [[note: a panel in this case can be an operable panel - which includes a frame and glazing. Thus the components breakdown will be:
Window < Lining + (Panels < Frames + Glazing + hardware) +Trim]]

Note: according to BSI 6100 - the LINING lines the opening (e.g. also called Jamb, Sill, Head), the FRAME is the friame immediatey around the door or window. Previously I had been calling these the "Frame" and "Inner Frame". Also, it should be noted that the work SASH means a sliding frame - a special type of frame. I have not made the distinction between fixed or operable

Resolution Change them for the Final.

Action # 1 ***Assignee*** Liebich ***Status*** Incomplete ***Resolved in Version*** R1.5 - Final

Make changes as proposed.

Issue Number	***I - 318***			***Issue Date***	11/26/97
Author	See	***Owner***	See	***Status***	Resolved
Schema	IfcProductExt	***Version***	R1.5 - Final Candi		

Issue Description IfcElement.ConnectedWith / IfcElement.ConnectedBy - the difference between these two is NOT CLEAR in the documentation - I am assuming that ConnectedWith is on the RelatingObject side and ConnectedBy is on the RelatedObject side, but it is NOT CLEAR from the documentation - NOR is it clear WHY this distinction is important (e.g. whay two sets of connections?).

Proposed Solution Rename to "ConnectedElements" and "ConnectionToElements" (clearer names) and add to documentation - RelatingObject/RelatedObject. The intent is to more clearly indicate the meaning behind the two lists. Since the RelatingObject side of an objectified relationship is intended to be the "driving" side of the relationship (if one side is driving), then this name is more 'possessive'.

Resolution Agreed.

Action #	1	***Assignee***	Liebich	***Status***	Complete	***Resolved in Version***	R1.5 - Final

make the changes as proposed

Issue Number	***I - 319***			***Issue Date***	11/16/97
Author	See	***Owner***	Liebich	***Status***	Resolved
Schema	IfcUtilityResource	***Version***	R1.5 - Final Candi		

Issue Description Hard to believe, but IfcTable still has some problems.

1) NoOfCellsInRow should be an attribute of IfcTableRow, NOT IfcTable. Because it is not used in the Table, but IS used in theTableRow (to set the length of the list of values). NOTE: this is currently INCORRECTLY referenced as "NoOfColumns" in the TableRow class.

2) NoOfHeadings and NoOfDataRows are inconsistently named.

Proposed Solution 1) move the attribute NoOfCellsInRow to the IfcTableRow class.

2) rename NoOfHeadings to NoOfHeadingRows

Resolution 1) Disagreed. Leaving NoOfCellsInRow as a derived value on the IfcTable provides an easy attribute that any app can check. NOTE: the documentation should be enhanced to clarify that the number of cells is DETERMINED by the number of cells in the first row and a WHERE rule insures that all other rows include the same number of cells. Attribute will be left on IfcTable.

2) No, want to avoid changes to the Schema for R1.5 addendum. This will be fixed in improved "Tables" design in R2.0.

Action #	1	***Assignee***	Liebich	***Status***	Complete	***Resolved in Version***	R1.5 - Addend

Enhance the documentation for IfcTable and IfcTableRow should be enhanced to clarify that the number of cells in all rows is DETERMINED by the number of cells in the first row and a WHERE rule on IfcTable insures that all other rows include the same number of cells

Issue Number	***I - 320***			***Issue Date***	11/26/97
Author	See	***Owner***	Wix	***Status***	Resolved
Schema	IfcDocumentExt	***Version***	R1.5 - Final Candi		

Issue Description Documentation for IfcCostScheduleGroup discusses the grouping of IfcCostScheduleElements -- yet this class does not exist in the schema. IfcRelCostScheduleElements is subtyped from IfcRelationship1toN, and points to a LIST of IfcProduct objects (as RelatedObjects), but they are not called IfcCostScheduleElements. It appears that the intent was --> IfcCostScheduleGroup groups IfcRelCostScheduleElements (but this cannot be done - IfcRelGroups groups IfcObjects (not IfcRelationships))

Further, IfcRelCostScheduleElement related IfcProduct objects directly to the IfcCostSchedule - seemingly bypassing the IfcCostScheduleGroup.

Proposed Solution 1) replace IfcRelCostScheduleElement with a subtype of IfcGroup called IfcCostElement (keeping all of the attributes currently defined - except the relationship to CostSchedule).
2) Document utilization of IfcRelGroups to group multiple objects into a single CostElement (note that this will be group of IfcObject rather than IfcProduct since we should allow costing of Process

and Proxy)
3) Document utilization of IfcRelGroups to group multiple IfcCostScheduleElements into a single CostScheduleGroup (as described in I-278).
4) Create a select type called IfcCostScheduleOrGroupSelect -- select for IfcCostScheduleGroup and IfcCostSchedule.
5) ReCreate (from PreFinal) objectified relationship called IfcRelGroupsCostSchedules (subtyped from IfcRelGroups) for which the RelatingObject is IfcCostSchedule and the LIST [1:?] of RelatedObjects are IfcCostScheduleOrGroupSelect

Resolution

1) agreed to create new class called IfcCostElement, but it is subtyped from IfcControl and is related to multiple IfcProducts through the IfcRelCostScheduleElements. This solves the "N to N" relationship problem in allowing a IfcProduct to be included in multiple IfcCostElements.
2) Disagreed - this is handled as described in (1) above.
3) This is done in the EXPRESS-G and in the documentation.
4) Agree to create the select type, but it will be referenced by the IfcCostSchedule only -- as the IfcRelGroups relationship will already allow us to "group" collections of IfcCostElementGroups and IfcCostElements.
5) This has been done as a simple relationship called "HasCostElementsOrGroups". We don't currently allow CostElements or CostElementGroups to be "part of" multiple Cost Schedules. This would appear to be a relationship that was missing from R1.5 FINAL and should be added for the Addendum.

Action # 1 ***Assignee*** Wix ***Status*** Complete ***Resolved in Version*** R1.5 - Final

Complete item (1) above

Action # 2 ***Assignee*** Wix ***Status*** Complete ***Resolved in Version*** R1.5 - Final

handle item (3) as described above.

Action # 3 ***Assignee*** Wix ***Status*** Complete ***Resolved in Version*** R1.5 - Final

complete item (4) as described above

Action # 5 ***Assignee*** Wix ***Status*** Incomplete ***Resolved in Version*** R1.5 - Final

complete item (5) as described above -- NOTE: the relationship from IfcCostSchedule ("HasCostElementsOrGroups") must be redirected to the select type "IfcCostElementOrGroupSelect" -- Add the missing relationship

Issue Number ***I - 321*** ***Issue Date*** 11/26/97

Author	See	***Owner***	Wix	***Status***	Deferred to R2.0
Schema	IfcPropertyResource	***Version***	R1.5 - Final Candi		

Issue Description

1) IfcClassificationNotation.NotationStrings -- these are more specifically called facets.
2) IfcClassificationNotation.Separator for each facet is too heavy
3) LIST [1:?] strings in a notation seems too heavy

Proposed Solution

1) "NotationStrings" --> should really be called "NotationFacets"
2) How about a single "separator" up on the ClassificationNotation object?
3) Probably want to limit the number of facets to 3 or 4. More than this becomes ridiculous (change using a WHERE rule)

(JW-980510) The proposed "C-Uni" model shows a proposed revised model of classification (using the Uniclass classification system as an example). This proposes a number of modifications that would enable us to use current classification systems directly within the IFC model. It is not yet fully complete. However, I believe it moves us towards a situation that would create a good set of common ground with classification specialists whilst providing additional flexibility over what we already have.

A key aspect of the revised model is that it introduces the notion of registered classification systems (IfcRegisteredClassificationEnum). A registered classification system is one that has created a hierarchical model that can be directly interpreted by an application to give the relevant classification information directly to a model that can be exported via IFC. Allowing that not all classification systems in existence will register (especially local or company systems) an IfcUnregisteredClassification is allowed that has a name and using which, a user would have to enter information directly. Selectionof registered or unregistered classification would be via an IfcClassificationSourceSelect select type.

The classification would have its edition and description as before. Description is an optional attribute. Edition is mandatory.

An IfcClassification can have a list of IfcClassicationFacet where each facet has attributes of table and notation (giving the value). The list of facets gives the potential for using multiple facets of a classification. We should not restrict the number of facets even though I agree that 3 or 4 is a sensible maximum; Uniclass has 11 tables and it is feasible (if impracticable) to use every one.

This gets rid of the NotationString class that was in the 1.5 model.

The key to populating the classification is in the provision of the classification hierarchy and we should encourage classification societies to do this. We have a number of such societies as members (NBS, Swedish organisation whose name I cannot pronounce, CSI etc.). Using these hierarchical models, it should be possible to populate the relevant attributes of the classification model. It will need some rules to achieve but I cannot see that it cannot be done. It would also stretch the capacity of the model significantly.

Note that items dealing with IfcClassificationList remain unchanged.

If this idea gains acceptance within the STF, I can float it further amongst classification specialists to see how they respond.

Resolution For R1.5 we will do 1 and 2.

For R2.0 we will discuss the proposal by Jeff

Action # 1 ***Assignee*** Wix ***Status*** Incomplete ***Resolved in Version*** R1.5 - Addend

change recommendations 1 and 2 for R1.5+1

Issue Number I - 322 ***Issue Date*** 12/9/97

Author NA Arch Group ***Owner*** Wix ***Status*** Deferred to R2.0

Schema IfcDocumentExt ***Version*** R1.5 - Final

Issue Description IfcMaterial -- Need to include a finish.

Proposed Solution Add an attribute "Finish : STRING"

Resolution (JW-980510) I would suggest that Finish is a separate class that should be applied to an element, is separate from the material, and is selected from a range of possible finishes. It could be an applied finish such as paint, and would have its own attributes such as emissivity, colour, reflectance – all of which are independent of material.

However, for R2, I have created the class IfcMaterialFinish with an optional HasFinish relation and an inverse AppliedTo relation that is a set since the same finish could be applied to many elements/materials. In this way, we do not have to create separate instances of IfcMaterial for every different type of Finish that might be applied which would otherwise be the case.

The Finish would also determine the surface spread of flame characteristics and so we should invite the AR2 team to contribute the extension requirements to this class for this purpose to provide more flexibility in the model and to enable its use within a domain process already established.

For the present, I have identified Color and FinishType as enumerations without attempting to fill out the lists. Architects, being creative beings, would probably want to use something like a Pantone list. There are probably other definitive lists around and so this might need to turn into a ColourRangeSelect in the longer term.

Action # 1 ***Assignee*** Wix ***Status*** Eliminated ***Resolved in Version*** R1.5 - Final

develop as described

Issue Number I - 323 ***Issue Date*** 12/9/97

Author NA PM Group ***Owner*** Wix ***Status*** Resolved

Schema IfcProcessExt ***Version*** R1.5 - Final

Issue Description We REALLY need to be able to use nested Processes (e.g. IfcWorkTask). That is, a WorkTask may (or may not) contain other WorkTasks, which may contain . . . The primary driver of this requirement is that different applications (e.g. cost estimating vs scheduling) will refer to different levels of these 'nesting trees' (e.g. estimating may 'cost' at the 3rd level of detail while scheduling may only 'schedule' at the 2nd level. This means that each of these applications must be able to 'manipulate' any level of these 'nesting trees' as a process object. Waiting until R2.0 (complete in Fall 1998) would cause hardship for Timberline and other cost estimating developers who are planning their development now. We would like to see a resolution completed in the R1.5 addendum.

Please see email thread between Tom Froese, Mike Cole, Kevin Yu and Richard See in early December.

Proposed Solution Enable nesting (recursive self references) in IfcWorkItem (NOTE: proposed renaming of IfcWorkTask). Note: this will eliminate IfcWorkGroup as the general purpose grouping mechanism does not work in this case.

Resolution Agreed in principle. Propose to solve this using general purpose solution allowing nesting of several subtypes of IfcObject -- see I-338 for solution description.

1) rename of IfcWorkTask to IfcWorkItem agreed (since the name "task" is relative to which level of a process hierarchy at which you look).

2) eliminate IfcWorkGroup as it will no longer be needed.

Action # 1 **Assignee** Wix **Status** Incomplete **Resolved in Version** R1.5 - Addend

Rename IfcWork to IfcWorkTask

Action # 2 **Assignee** Wix **Status** Incomplete **Resolved in Version** R1.5 - Addend

Eliminate IfcWorkGroup as it will no longer be needed (replaced by nesting) and insure adaptation of Process schema to take advantage of the general purpose solution provided by I-338

Action # 3 **Assignee** Wix **Status** Incomplete **Resolved in Version** R1.5 - Addend

Insure that the general purpose solution provided by I-338 will satisfy the requirements of the issue listed above.

Action # 4 **Assignee** Wix **Status** Incomplete **Resolved in Version** R1.5 - Addend

Enhance documentation for IfcProcess (and/or IfcWorkTask) to insure that the reader understands how to make use of the general purpose nesting mechanism (I-338).

Issue Number **I - 324** **Issue Date** 12/9/97

Author	NA PM Group	**Owner**	Wix	**Status**	Resolved
Schema	IfcKernel	**Version**	R1.5 - Final		

Issue Description In Release 1.0 we were able to "Type" IfcResource as one of "Labor", "Equipment" or "Material". This has been removed from R1.5 and should not have been. We need it back. See email thread from early December 1997.

Proposed Solution Add the GenericType and other reasonable attributes (that were included in IFC R1.0) back onto IfcResource.

Resolution Agreed

Action # 1 **Assignee** Liebich **Status** Incomplete **Resolved in Version** R1.5 - Addend

enhance IfcResource as described in the proposed solution above.

Issue Number **I - 325** **Issue Date** 12/9/97

Author	NA PM Group	**Owner**	Wix	**Status**	Resolved
Schema	IfcKernel	**Version**	R1.5 - Final		

Issue Description We (estimators and schedulers) need to be able to use nested Resources (IfcResource). For example, it is quite common in an estimate or schedule to list a work crew or subcontractor as a resource for complex tasks or sub-processes. Such a 'crew' will be bid at a set rate per hour or per day - which is what should be included in an estimate - at the 'crew' level.

Proposed Solution Enable nesting (recursive self referencing) in IfcResource.

Resolution Agreed in principle. Propose to solve this using general purpose solution allowing nesting of several subtypes of IfcObject -- see I-338 for solution description.

Action # 1 **Assignee** Wix **Status** Incomplete **Resolved in Version** R1.5 - Addend

Insure that the general purpose solution provided by I-338 will satisfy the requirements of the issue listed above.

Action # 2 **Assignee** Wix **Status** Incomplete **Resolved in Version** R1.5 - Addend

Enhance documentation for IfcResource to insure that the reader understands how to make use of the general purpose nesting mechanism (I-338).

Issue Number ***I - 326*** **Issue Date** 12/9/97

Author NA PM Group **Owner** Wix **Status** Resolved

Schema IfcPropertyResource **Version** R1.5 - Final

Issue Description We (estimators and schedulers) need to be able to use nested cost elements (IfcCostElement). That is, a cost element may contain other cost elements . . . The reason is that estimates are prepared at various levels of detail. A cost element in one estimate may be a hierarchy (or nested) set of cost elements in another estimate. It is not practical to maintain different estimate hierarchies for these. We need to be able to 'use' different levels of detail, knowing that each contains (and sums) all of the lower level contained CostElements.

Proposed Solution Enable nesting (recursive self referencing) in IfcCostElement.

Resolution Agreed in principle. Propose to solve this using general purpose solution allowing nesting of several subtypes of IfcObject -- see I-338 for solution description.

1) eliminate IfcWorkGroup as it will no longer be needed.

Action # 1 **Assignee** Wix **Status** Complete **Resolved in Version** R1.5 - Addend

Eliminate IfcWorkGroup and insure that the general purpose solution provided by I-338 will satisfy the requirements described for this issue.

Action # 2 **Assignee** Wix **Status** Incomplete **Resolved in Version** R1.5 - Addend

Insure that the general purpose solution provided by I-338 will satisfy the requirements of the issue listed above.

Action # 3 **Assignee** Wix **Status** Incomplete **Resolved in Version** R1.5 - Addend

Enhance documentation for IfcCostElement to insure that the reader understands how to make use of the general purpose nesting mechanism (I-338).

Issue Number ***I - 327*** **Issue Date** 12/12/97

Author See **Owner** Wix **Status** Resolved

Schema IfcPropertyResource **Version** R1.5 - Final

Issue Description Why are IfcMaterialLayer and IfcMaterialLayerSetUsage NOT subtyped from IfcProperty when all of the other classes related to materials are?? (e.g. IfcMaterial, IfcMaterialLayerSet, IfcMaterialList)

Proposed Solution Subtype from IfcProperty (?)

Resolution Agreed

Action # 1 **Assignee** Wix **Status** Incomplete **Resolved in Version** R1.5 - Addend

Change for R1.5 Addendum

Issue Number ***I - 328*** **Issue Date** 12/12/97

Author See **Owner** Liebich **Status** Resolved

Schema IfcPropertyResource **Version** R1.5 - Final

Issue Description IfcProjectMaterialRegistry should be defined at at the Kernel level, as with the other registrys related to the Project. This will allow an inverse relationship from this registry to the Project -- as with the other registrys

Proposed Solution 1) Move IfcProjectMaterialRegistry into the Kernel so that it can be referenced by Project
2) add an inverse relationship from IfcProjectMaterialRegistry to IfcProject as with the other registries

Resolution 1) Don't need to move it to the Kernel. IfcProject can reference it within the IfcPropertyResource in the same way as it references the other two registries in the IfcUtilitiesResource.

2) Don't need the inverse relationship for R1.5 -- consider a general purpose Project Registry for R2.0 - defined at the Kernel level.

Action # 1 **Assignee** Liebich **Status** Incomplete **Resolved in Version** R1.5 - Addend

Add a reference from IfcProject (in IfcKernel) -- called ProjectMaterialRegistry : IfcMaterialRegistry (same as the references to the other two registries on IfcProject).

Issue Number	**I - 329**			**Issue Date**	12/10/97
Author	Forester	**Owner**	See	**Status**	Resolved
Schema	IfcSharedBldgElements	**Version**	R1.5 - Final		

Issue Description There is currently no way to tell if an occurrence of IfcWall is "interior" or "exterior". This is critical for thermal performance simulation and thermal load calculation applications.

Proposed Solution Add an "Exterior" property to the Pset_WallType (common to all Walls) which is type IfcBoolean.

Resolution Agreed

Action # 1 **Assignee** See **Status** Complete **Resolved in Version** R1.5 - Addend

make the addition to the Pset_WallType property set

Issue Number	**I - 330**			**Issue Date**	12/10/97
Author	Autodesk reviewers	**Owner**	See	**Status**	Resolved
Schema	IfcPropertyTypeResource	**Version**	R1.5 - Final		

Issue Description An IfcProductShape has an IfcProductComponentShape.

An IfcProductComponentShape is either an IfcShapeBody or an IfcShapeResult

An IfcShapeBody contains a list of IfcShapeRepresentations

My understanding is that this is to allow for multiple representations of an object. For example, there is always a bounding box, and there might be different geometric representations for different kinds of views.

This next part is where I get confused:

an IfcShapeResult is basically a boolean of two or more IfcProductComponentShapes

This means that an IfcShapeResult can be a boolean of two IfcShapeBodies, but IfcShapeBody is the thing that has multiple representations. How are you supposed to boolean together the sets of multiple representations? It seems to me that the IfcShapeResult is at too high a level.

Proposed Solution 1) Move the componentization concept down to IfcShapeRepresentation level so that the componentization of a representation is done at the Representation level --> this will allow such componentization to be different for each representation. See proposed alternative "ShpR_new.exg"

Resolution Have discussed two alternatives to solving this for R1.5 addendum:
1) severely limit the Product Shape schema --> single shape representation allowed
2) implement proposed longer term solution early

Agreed that we will implement #2 for R1.5 addendum

Action # 1 **Assignee** Liebich **Status** Incomplete **Resolved in Version** R1.5 - Addend

complete proposal and send to RS to incorporate in IfcPropertyTypeResource schema

Action # 2 **Assignee** See **Status** Incomplete **Resolved in Version** R1.5 - Addend

Incorporate solution develoed by T.Liebich into the IfcPropertyTypeResource schema

Issue Number **I - 331** **Issue Date** 1/9/98

Author	Haiat	***Owner***	Liebich	***Status***	Resolved
Schema	IfcGeometryResource	***Version***	R1.5 - Final		

Issue Description In EXPRESS the Range of Attribute ClippingHalfSpaces for IfcAttDrivenClippedExtudedSolid and IfcAttDrivenClippedRevolvedSolid is contraint to [1:2], whereas Express-G and Specs show [1:?].

Proposed Solution The [1:?] is correct and shall be updated in EXPRESS.

Resolution Agreed

Action #	0	***Assignee***	Liebich	***Status***	Complete	***Resolved in Version***	R1.5 - Addend

"Just do it" (TL)

Issue Number I - 332 ***Issue Date*** 1/9/98

Author	Horvath, Jens-Peter	***Owner***	Liebich	***Status***	Resolved
Schema	IfcGeometryResource	***Version***	R1.5 - Final		

Issue Description At IfcAxis2Placement3D: It is not clear from the Specification, that the default for Attribute RefDirection is [1.0,0.0,0.0].

Proposed Solution Update the documentation.

Resolution Agreed

Action #	1	***Assignee***	Liebich	***Status***	Complete	***Resolved in Version***	R1.5 - Addend

make change to documentation

Issue Number I - 333 ***Issue Date*** 1/9/98

Author	Horvath, Jens-Peter	***Owner***	Liebich	***Status***	Resolved
Schema	IfcGeometryResource	***Version***	R1.5 - Final		

Issue Description At IfcCurveBoundedPlane the default and the min value for Dim shall be 3, not 2.

Proposed Solution Update the documentation

Resolution Agreed

Action #	1	***Assignee***	Liebich	***Status***	Complete	***Resolved in Version***	R1.5 - Final

"Just do it"

Issue Number I - 334 ***Issue Date*** 2/1/98

Author	Ohta, Takakazu	***Owner***	Liebich	***Status***	Resolved
Schema	IfcProductExt	***Version***	R1.5 - Final		

Issue Description At IfcBuildingStorey: The specification shows for calcTotalArea the data type IfcLenghtMeasure, the correct data type is IfcAreaMeasure. (express and express-g are correct).

Proposed Solution change specification

Resolution Agreed

Action #	1	***Assignee***	Liebich	***Status***	Incomplete	***Resolved in Version***	R1.5 - Addend

change specification as described

Issue Number I - 335 ***Issue Date*** 2/1/98

Author	Forester	***Owner***	Liebich	***Status***	Resolved
Schema	IfcMeasureResource	***Version***	R1.5 - Final		

Issue Description IfcMeasureValue currently does not include the IfcString, IfcBoolean, IfcInteger, IfcReal in its select list within the EXPRESS code view of the model (EXPRESS-G and Specification are correct).

Proposed Solution Correct the EXPRESS code to add these four types to the select type.

Resolution Agreed

Action # 1 **Assignee** Wix **Status** Incomplete **Resolved in Version** R1.5 - Addend

Change the EXPRESS code

Issue Number ***I - 336*** **Issue Date** 2/1/98

Author Muigg **Owner** Liebich **Status** Resolved

Schema IfcProductExt **Version** R1.5 - Final

Issue Description Support for logical connections between elements has been disabled between the R1.5 Pre-Final and the R1.5 Final versions of the model. IfcRelConnectsElements a now has an attribute called ConnectionGeometry WHICH IS MANDATORY. This means that the application MUST provide connection geometry and logical connections of path based elements (in which the connection location is calculated by the app) are disabled. Implementers CLEARLY wanted to include support for logical connection of Path based elements.

Proposed Solution Change the ConnectionGeometry attribute on IfcRelConnectsElements to be OPTIONAL

Resolution Agreed

Action # 1 **Assignee** Liebich **Status** Incomplete **Resolved in Version** R1.5 - Addend

change the IfcProductExt schema accordingly

Issue Number ***I - 337*** **Issue Date** 2/1/98

Author See **Owner** See **Status** Deferred to R2.0

Schema IfcDocumentExt **Version** R1.5 - Final

Issue Description IfcCostElement is NOT a control -- it is more like an Aspect (or data view) of other objects.

Proposed Solution Subtype from IfcObject for R1.5 (since there is nothing added in IfcControl now anyway and this reduces the depth in the hierarchy) and subtype from IfcAspect in R2.0.

Resolution Agreed that it is not a control. However, subtyping from IfcObject is not a good idea (bad precedent). Leave it where it is for R1.5 addendum and look again under the IfcAspect proposed for R2.0 (BS-4 project).

Issue Number ***I - 338*** **Issue Date** 2/9/98

Author Liebich **Owner** See **Status** Resolved

Schema IfcKernel **Version** R1.5 - Final

Issue Description Issues I-323 (Processes), I-325 (Resources) and I-326 (Cost Elements) -- all describe the requirement for nesting in primary subtypes of IfcObject. This was also the case with I-106 (nesting of IfcBuildingElements (Ifcproducts)). It will be inefficient to define 4 different (or redundant) solutions.

Proposed Solution Consider defining an objectified relationship at the IfcObject level that will allow nesting of like type elements (to be checked by a WHERE rule). See diagram "GeneralGrouping.vsd"

Resolution Implement as described in "GeneralGrouping" proposal. See notes on I-323, I-325, I-326 for cleanup of old solutions and checking that new solution works as well.

Decided that we cannot remove IfcRelAssemblesElements because it allows assembly of dissimilar element types.

Action # 1 **Assignee** Liebich **Status** Incomplete **Resolved in Version** R1.5 - Addend

Make the addition to IfcObject -- as described above

Action # 2 **Assignee** Liebich **Status** Eliminated **Resolved in Version** R1.5 - Addend

Remove IfcRelAssemblesElements in the IfcProductExt schema and replace with a note explaining how this is now covered by the general purpose solution added at the IfcObject level.

Decided that we cannot remove IfcRelAssemblesElements because it allows assembly of dissimilar element types.

Action # 3 ***Assignee*** See ***Status*** Complete ***Resolved in Version*** R1.5 - Addend

note on I-323 (Processes), I-325 (Resources) and I-326 (Cost Elements) that this general purpose solution addresses those requirements and ADD NEW ACTIONS to enhance documention which describes this.

Issue Number ***I - 339*** ***Issue Date*** 9/4/98

Author Han, Chuck ***Owner*** See ***Status*** Resolved

Schema All Schemata ***Version*** R1.5 - Final

Issue Description EXPRESS allows you to redeclare the data type for attributes in subtype classes. IDL does not. This creates a problem in developing IDL code that is consistent with the EXPRESS.

Proposed Solution See if it is possible to avoid redeclaration of attribute data types. See also the issue logged by Tim Child regarding the Von Liskov principal in OO design.

Resolution Assumption: the only place we have done this is in redeclaring relationships on Obj. Rels.

If this is true, then this is resolved by the resolution to I-310.

Issue Number ***I - 340*** ***Issue Date*** 2/4/98

Author Han, Chuck ***Owner*** See ***Status*** Resolved

Schema All Schemata ***Version*** R1.5 - Final

Issue Description IDL compilers tested complained about duplicate names in Enumerations.

Proposed Solution Eliminate duplicate names by prepending the name of the class or something similar. This should probably also make EXPRESS compilers happier

Resolution Agreed -- will preface the enum values with the name of the enumeration as is done automatically by EXPRESS compilers.

Action # 1 ***Assignee*** Hietanen ***Status*** Incomplete ***Resolved in Version*** R1.5 - Addend

make the change to the IDL generation

Issue Number ***I - 341*** ***Issue Date*** 3/11/98

Author Bouman-Eijs, Anita ***Owner*** See ***Status*** Resolved

Schema All Schemata ***Version*** R1.5 - Final

Issue Description The following errors are reported by the EPM EXPRESS compiler:

```
> ----- Errors and warning in Ifc150_Final_Express_LF.exp -----
> ENTITY IfcRevolvedAreaSolid
> In the assignment of derived attribute AxisLine, the entity
> constructor
> of supertype IfcCurve is missing.
>
> FUNCTION IfcCircleProfileIntoCurve
> In the assignment of local variable Circle, the constructor of
> supertype
>
> IfcCurve is missing.
> In the assignment of local variable ResCurve, the constructor of
> supertype IfcCurve is missing.
>
> FUNCTION IfcRectangleProfileIntoCurve
> In the assignment of local variable ResCurve, the constructors of
> supertype IfcBoundedCurve and IfcCurve are missing.
>
> FUNCTION IfcTrapeziumProfileIntoCurve
> In the declaration of local variable TempPoint, the constructor of
> supertype IfcPoint is missing.
> In the assignment of local variable ResCurve, the constructors of
> supertype IfcBoundedCurve and IfcCurve are missing.
>
> FUNCTION IfcPointTranslation
> In the assignment of local variable Point, the constructor of
```

> supertype
>
> IfcPoint is missing.
>
> FUNCTION IfcRevolutionPath
> In the declaration of local variable Circle, the constructor of
> supertype IfcCurve is missing.
> In the assignment of local variable Path, the constructor of
> supertype
> IfcCurve is missing.
>
> FUNCTION IfcProfileIntoArea
> In the assignment of local variable ResSurface, the constructor of
> supertype IfcPoint is missing.
>
> ENTITY IfcExtrudedAreaSolid
> The supertype clause to entity IfcAttDrivenExtrudedSegment is missing.
> (Warning)
>
> ----- Error in IfcDocumentExtension.exp -----
> In REFERENCE clause to schema IfcKernel are IfcProduct and IfcControl
> missing.
>
> ----- Error in IfcKernel.exp -----
> In REFERENCE clause to schema IfcUtilityResource are
> IfcProjectTeamRegistry and IfcProjectAppRegistry missing.
>
> ----- Error in IfcModelingAidExtension.exp -----
> In REFERENCE clause to schema IfcGeometryResource is IfcBoundedCurve
> missing.
>
> ----- Error in IfcProcessExtension.exp -----
> In REFERENCE clause to schema IfcPropertyResource is IfcDateTimeSelect
> missing.
>
> ----- Errors in IfcProductExtension.exp -----
> In USE clause to schema IfcKernel is IfcControl missing.
> In REFERENCE clause to schema IfcMeasureResource is
> IfcPositiveLengthMeasure missing.
>
> ----- Error in IfcUtilityResource.exp -----
> In REFERENCE clause to schema IfcMeasureResource is IfcMeasureValue
> missing.
> ----- end -----

Proposed Solution Resolve each EXPRESS error in turn

Resolution Agreed -- method to be determined.

Action # 1 ***Assignee*** Liebich ***Status*** Incomplete ***Resolved in Version*** R1.5 - Addend

Resolve EXPRESS compiler errors for Addendum

Issue Number I - 342 ***Issue Date*** 3/12/98

Author Liebich ***Owner*** Wix ***Status*** Resolved

Schema IfcPropertyResource ***Version*** R1.5 - Final

Issue Description On IfcMaterial -- the attribute MaterialClassification is mandatory. That means, we always require the
classification of material in an IFC file/db.

Proposed Solution My proposal would be to make MaterialClassification optional.

Resolution Agreed

Action # 1 ***Assignee*** Wix ***Status*** Incomplete ***Resolved in Version*** R1.5 - Addend

Make the change as proposed

Issue Number I - 343 ***Issue Date*** 3/18/98

Author Liebich ***Owner*** See ***Status*** Resolved

Schema IfcProductExt ***Version*** R1.5 - Final

Issue Description Class: IfcBuilding

The inverse for IfcRelContains on this class [xxx] does not limit the container to IfcSite object. This is a problem since IFC model integrity assumes the containment hierarchy --> site -> building -> building storey -> space

Proposed Solution Add a second WHERE rule :

WR2: SIZEOF(QUERY(Temp <* IsContainedBy |
Temp.RelationshipType = SiteContainer)) = 1;

Resolution Agreed

Action # 1 ***Assignee*** Liebich ***Status*** Incomplete ***Resolved in Version*** R1.5 - Addend

"Just do it"

Issue Number ***I - 344*** ***Issue Date*** 3/18/98

Author Liebich ***Owner*** See ***Status*** Resolved

Schema IfcGeometryResource ***Version*** R1.5 - Final

Issue Description Class: IfcAxis2Placement2D

Currently there is no contraint that prohibits the use of a three dimensional points for the location of a two dimensional placement

Proposed Solution add a second WHERE rule:

WR2: SELF\IfcPlacement.Location.Dim=2;

Resolution Agreed

Action # 1 ***Assignee*** Liebich ***Status*** Complete ***Resolved in Version*** R1.5 - Addend

"Just do it"

Issue Number ***I - 345*** ***Issue Date*** 4/25/98

Author See ***Owner*** See ***Status*** Deferred to R2.0

Schema All Schemata ***Version*** R1.5 - Final

Issue Description We need a method to automate the generation of EXG files (from EXPRESS).

Proposed Solution Use EDM tools for this

Resolution

1. Will use EDM for automated generation of EXG files.
2. Will ask VTT about purchase of a license for EDM and about providing experts to generate the EXG files through the development of R2.0.
3. Note: will try to find a method for adding notes on redeclared relationships (as we do on subtyped objectified relationships) in order to clarify the semantic meaning of the redeclared relationship.

Action # 1 ***Assignee*** Hyvarinen ***Status*** Incomplete ***Resolved in Version*** R2.0 - Beta

Complete initial testing with EDM and document process for semi-automated generation of EXG diagrams from EXPRESS. Also want to check the STEP TOOLS EXG generation.

Action # 2 ***Assignee*** Hyvarinen ***Status*** Incomplete ***Resolved in Version*** R2.0 - Beta

Follow through with VTT about purchase and completing the EXG generation through the R2.0 project.

Issue Number ***I - 346*** ***Issue Date*** 5/5/98

Author See ***Owner*** Liebich ***Status*** Resolved

Schema IfcProductExt ***Version*** R1.5 - Final

Issue Description 1) IfcRelAssemblesSpaces.RelatedObjects - this should be a LIST [0:?] IfcSpace. Currently it is a single IfcSpace -- which breaks the interface contract extablished in the Supertype IfcReleationship1toN.

2) The name for this class is misleading. The original intent was to allow nesting of spaces. The name implies assembling (grouping) which is different.

Proposed Solution 1) Change IfcRelAssemblesSpaces.RelatedObjects to a LIST [0:?] IfcSpace

2) change the name to IfcRelNestsSpaces

Resolution Agreed. However, see solutions to I-323, I-325 and I-326. If a general purpose solution is used at the IfcObject level, this objectified relationship may be removed because it will be redundant with such a general purpose nesting solution.

Action # 1 ***Assignee*** Liebich ***Status*** Incomplete ***Resolved in Version*** R1.5 - Final

Change as described.

Issue Number I - 347 ***Issue Date*** 5/5/98

Author Monceyron ***Owner*** Liebich ***Status*** Resolved

Schema All Schemata ***Version*** R1.5 - Final

Issue Description The following issues with WHERE rules have been identified within CSTB:

```
******************************

// Issue with WR2: validation always returns False
// IfcMaterial type is not a selection item of IfcMaterialSelect select type

ENTITY IfcColumn
 SUBTYPE OF (IfcBuildingElement);
   GenericType : IfcColumnTypeEnum;
 WHERE
   WR1: SIZEOF(QUERY( Temp <* SELF\IfcObject.TypeDefinitions |
         NOT(Temp.TypedClass = 'IfcColumn'))) = 0;
   WR2: 'IFC150FINAL.IFCMATERIAL' IN TYPEOF(SELF\IfcBuildingElement.HasMaterial);
END_ENTITY;

TYPE IfcMaterialSelect = SELECT (
   IfcMaterialLayerSet
  ,IfcMaterialList);
 END_TYPE

******************************

// Issue with WR2: validation always returns False
// IfcMaterial type is not a selection item of IfcMaterialSelect select type

ENTITY IfcBeam
 SUBTYPE OF (IfcBuildingElement);
   GenericType : IfcBeamTypeEnum;
 WHERE
   WR1: SIZEOF(QUERY( Temp <* SELF\IfcObject.TypeDefinitions |
         NOT(Temp.TypedClass = 'IfcBeam'))) = 0;
   WR2: 'IFC150FINAL.IFCMATERIAL' IN TYPEOF(SELF\IfcBuildingElement.HasMaterial);
END_ENTITY;

TYPE IfcMaterialSelect = SELECT (
   IfcMaterialLayerSet
  ,IfcMaterialList);
 END_TYPE

******************************

ENTITY IfcAttDrivenMorphedExtrudedSegment
 SUBTYPE OF (IfcAttDrivenExtrudedSegment);
   EndProfileDef   : IfcAttDrivenProfileDef;
 DERIVE
   EndSweptArea    : IfcCurveBoundedPlane
```

```
                := IfcProfileIntoArea(EndProfileDef);
 WHERE
    WR1: TYPEOF(SELF\IfcAttDrivenExtrudedSegment.ProfileDef) = TYPEOF(EndProfileDef);
    WR2: NOT('IFC150FINAL.IFCARBITRARYPROFILEDEF' IN
TYPEOF(SELF\IfcAttDrivenRevolvedSegment.ProfileDef));
    WR3: SELF\IfcAttDrivenExtrudedSegment.ProfileDef.Position.P[1] =
EndProfileDef.Position.P[1];
END_ENTITY;
```

An issue with WR2 : IfcAttDrivenRevolvedSegment is not a subtype of IfcAttDrivenMorphedExtrudedSegment
Thus, specification SELF\IfcAttDrivenRevolvedSegment.ProfileDef is wrong.
A guess could be : SELF\IfcAttDrivenExtrudedSegment.ProfileDef

****

```
ENTITY IfcAttDrivenMorphedExtrudedSegment
 SUBTYPE OF (IfcAttDrivenExtrudedSegment);
    EndProfileDef   : IfcAttDrivenProfileDef;
 DERIVE
    EndSweptArea    : IfcCurveBoundedPlane
                := IfcProfileIntoArea(EndProfileDef);
 WHERE
    WR1: TYPEOF(SELF\IfcAttDrivenExtrudedSegment.ProfileDef) = TYPEOF(EndProfileDef);
    WR2: NOT('IFC150FINAL.IFCARBITRARYPROFILEDEF' IN
TYPEOF(SELF\IfcAttDrivenRevolvedSegment.ProfileDef));
    WR3: SELF\IfcAttDrivenExtrudedSegment.ProfileDef.Position.P[1] =
EndProfileDef.Position.P[1];
END_ENTITY;
```

An issue with WR3: is at stake to test equality between two instances of IfcDirection ?
Should we test an equality member to member or an equality of directions - with a geometric meaning ?

The same kind of problem is encoutered with entity IfcAttDrivenExtrudedSolid

```
ENTITY IfcAttDrivenExtrudedSolid
 SUPERTYPE OF (ONEOF (
    IfcAttDrivenClippedExtrudedSolid))
 SUBTYPE OF (IfcSolidModel);
    Segments        : LIST [1:?] OF IfcAttDrivenExtrudedSegment;
 DERIVE
    Path            : IfcPolyline := IfcExtrusionPath(SELF);
 WHERE
    WR1: SIZEOF(QUERY( Temp <* Segments | Temp.Position.Axis <>
Segments[1].Position.Axis)) = 0;
END_ENTITY;
```

****

```
ENTITY IfcAttDrivenRevolvedSegment
 SUPERTYPE OF
(ONEOF(IfcAttDrivenMorphedRevolvedSegment,IfcAttDrivenTaperedRevolvedSegment))
 SUBTYPE OF (IfcRevolvedAreaSolid);
    Position        : IfcAxis2Placement3D;
    StartAngle      : IfcPlaneAngleMeasure;
    ProfileDef      : IfcAttDrivenProfileDef;
 DERIVE
    SELF\IfcSweptAreaSolid.SweptArea : IfcCurveBoundedPlane
                := IfcProfileIntoArea(ProfileDef);
 INVERSE
    PartOfSolid     : IfcAttDrivenRevolvedSolid FOR Segments;
 WHERE
    WR1: SELF\IfcRevolvedAreaSolid.Axis.Location.Coordinates[3] = 0;
END_ENTITY;
```

Issue with WR1: third element of Coordinates may not exist as
Coordinates : LIST [1:3] OF IfcLengthMeasure

****

```
ENTITY IfcArbitraryProfileDef
 SUBTYPE OF (IfcAttDrivenProfileDef);
   CurveForSurface : IfcBoundedCurve;
 WHERE
   WR1: (('IFC150FINAL.IFCPOLYLINE' IN
           TYPEOF(CurveForSurface)) AND (CurveForSurface.Dim = 2))
        OR
        (('IFC150FINAL.IFCTRIMMEDCURVE' IN
           TYPEOF(CurveForSurface)) AND (CurveForSurface.Dim = 2))
        OR
        (('IFC150FINAL.IFCCOMPOSITECURVE' IN
           TYPEOF(CurveForSurface)) AND (CurveForSurface.Dim = 2));
END_ENTITY;
```

issue with WR1 : attribute Dim is not defined at the level of IfcBoundedCurve but within each subtype of IfcBoundedCurv.

```
ENTITY IfcRelContains
 SUBTYPE OF (IfcRelationship1toN);
   RelationshipType       : IfcContainmentTypeEnum;
   ContainedOrReferenced  : BOOLEAN;
 WHERE
   WR1: ((RelationshipType = ProjectContainer) AND
         ('IFC150FINAL.IFCPROJECT' IN TYPEOF(SELF\IfcRelationship1toN.RelatingObject)))
        XOR (RelationshipType <> ProjectContainer);
   WR2: ((RelationshipType = SiteContainer) AND
         ('IFC150FINAL.IFCSITE' IN TYPEOF(SELF\IfcRelationship1toN.RelatingObject)) AND
         NOT('IFC150FINAL.IFCPROJECT' IN
TYPEOF(SELF\IfcRelationship1toN.RelatedObjects)))
        XOR (RelationshipType <> SiteContainer);
   WR3: ((RelationshipType = BuildingContainer) AND
         ('IFC150FINAL.IFCBUILDING' IN TYPEOF(SELF\IfcRelationship1toN.RelatingObject)) AND
         NOT('IFC150FINAL.IFCPROJECT' IN TYPEOF(SELF\IfcRelationship1toN.RelatedObjects))
AND
         NOT('IFC150FINAL.IFCSITE' IN TYPEOF(SELF\IfcRelationship1toN.RelatedObjects)))
        XOR (RelationshipType <> BuildingContainer);
   WR4: ((RelationshipType = BuildingStoreyContainer) AND
         ('IFC150FINAL.IFCBUILDINGSTOREY' IN
TYPEOF(SELF\IfcRelationship1toN.RelatingObject)) AND
         NOT('IFC150FINAL.IFCPROJECT' IN TYPEOF(SELF\IfcRelationship1toN.RelatedObjects))
AND
         NOT('IFC150FINAL.IFCSITE' IN TYPEOF(SELF\IfcRelationship1toN.RelatedObjects)) AND
         NOT('IFC150FINAL.IFCBUILDING' IN
TYPEOF(SELF\IfcRelationship1toN.RelatedObjects)))
        XOR (RelationshipType <> BuildingStoreyContainer);
   WR5: ((RelationshipType = SpaceContainer) AND
         ('IFC150FINAL.IFCSPACE' IN TYPEOF(SELF\IfcRelationship1toN.RelatingObject)) AND
         NOT('IFC150FINAL.IFCPROJECT' IN TYPEOF(SELF\IfcRelationship1toN.RelatedObjects))
AND
         NOT('IFC150FINAL.IFCSITE' IN TYPEOF(SELF\IfcRelationship1toN.RelatedObjects)) AND
         NOT('IFC150FINAL.IFCBUILDING' IN TYPEOF(SELF\IfcRelationship1toN.RelatedObjects))
AND
         NOT('IFC150FINAL.IFCBUILDINGSTOREY' IN
TYPEOF(SELF\IfcRelationship1toN.RelatedObjects)))
        XOR (RelationshipType <> SpaceContainer);
END_ENTITY;
```

Issue : the type ofSELF\IfcRelationship1toN.RelatedObjects is a list of IfcObject
(TYPEOF(SELF\IfcRelationship1toN.RelatedObjects))=LIST)
and then the test will fail

****
****

Proposed Solution see comments in the text above

Resolution Agreed - mostly -- TL will work with CSTB to find agreement.

Action # 1 ***Assignee*** Liebich ***Status*** Incomplete ***Resolved in Version*** R1.5 - Addend

Work w/ CSTB expert to resolve all

Issue Number ***I - 348*** ***Issue Date*** 5/10/98

Author Liebich ***Owner*** Wix ***Status*** Resolved

Schema IfcPropertyResource ***Version*** R1.5 - Final

Issue Description IfcMaterialList is not a list of materials in EXPRESS, since currently the attribute Materials is a single attribute.

Proposed Solution Update EXPRESS so that IfcMaterialList.Materials is a List [1:?]

Resolution Agreed

Action # 1 ***Assignee*** Wix ***Status*** Incomplete ***Resolved in Version*** R1.5 - Final

Update EXPRESS schema as proposed

Issue Number ***I - 349*** ***Issue Date*** 5/8/98

Author Liebich ***Owner*** See ***Status*** Resolved

Schema All Schemata ***Version*** R1.5 - Final

Issue Description "TypeDescription" fields described in many of the Pset definitions is really an attribute of the IfcPropertySet object. It is NOT one of the LIST [1:?] OF IfcProperty.

Proposed Solution It should be clearly separated in the spreadsheet definitions.

Resolution Actually, this is not true. The "Descriptor" attribute on IfcPropertySet should really be renamed to "PsetName" -- and should contain the name of the Pset (from the definition spreadsheets). For example, "Pset_DoorSliding". Therefore, the "TypeDescription" property is still needed to capture the user description for this type (e.g. "Pella 8' sliding door").

Action # 1 ***Assignee*** Liebich ***Status*** Complete ***Resolved in Version*** R1.5 - Final

change the name of the attribute on IfcPropertySet from "Descriptor" to "PsetName" to more accurately reflect the purpose of this attribute.

Issue Number ***I - 350*** ***Issue Date*** 5/8/98

Author Liebich ***Owner*** See ***Status*** Resolved

Schema All Schemata ***Version*** R1.5 - Final

Issue Description All references to nested Psets (inside other Psets) are currently shown as IfcObjectReference(s). This is not necessary since IfcPropertySet is a subtype of IfcProperty -- and can therefore be referenced directly.

Proposed Solution They should all be changed to the data type IfcPropertySet.

Resolution This is only true in the case of references to IfcSharedPropertySet (where there is a 1 to 1 relationship). Referenced to IfcOccurrencePropertySet should be handled as described in I-306

Action # 1 ***Assignee*** See ***Status*** Complete ***Resolved in Version*** R1.5 - Addend

Update all references for IfcSharedPropertysets defined in R1.5 (for which you are responsible)

Action # 2 ***Assignee*** Forester ***Status*** Incomplete ***Resolved in Version*** R1.5 - Addend

Update all references for IfcSharedPropertysets defined in R1.5 (for which you are responsible)

Action # 3 ***Assignee*** Yu ***Status*** Incomplete ***Resolved in Version*** R1.5 - Addend

Update all references for IfcSharedPropertysets defined in R1.5 (for which you are responsible)

Issue Number ***I - 351*** ***Issue Date*** 5/14/98

Author Liebich ***Owner*** Liebich ***Status*** Resolved

Schema IfcSharedBldgElements ***Version*** R1.5 - Final

Issue Description the WR2 at IfcBeam and IfcColumn is wrong, since it states, that the material information has to be of type IfcMaterial:
WR2: 'IFCPROPERTYRESOURCE.IFCMATERIAL' IN TYPEOF(SELF\IfcBuildingElement.HasMaterial);
However IfcMaterial is not a member of IfcMaterialSelect, the attribute type of HasMaterial

Proposed Solution Change WR so that it requests IfcMaterialList as type.
WR2: 'IFCPROPERTYRESOURCE.IFCMATERIALLIST' IN TYPEOF(SELF\IfcBuildingElement.HasMaterial);

Resolution Agreed

Action # 1 ***Assignee*** Liebich ***Status*** Incomplete ***Resolved in Version*** R1.5 - Addend

Change the WR as described

Issue Number I - 352 ***Issue Date*** 4/30/98

Author Drogemuller ***Owner*** Wix ***Status*** Deferred to R2.0

Schema IfcPropertyResource ***Version*** R1.5 - Final

Issue Description A layered material may need to be stored as part of a layered building element.

Proposed Solution Allow recursive references in MaterialLayerSets -- allow a layer to be a layer set.

Resolution

Issue Number I - 353 ***Issue Date*** 4/30/98

Author Drogemuller ***Owner*** Liebich ***Status*** Deferred to R2.0

Schema IfcProductExt ***Version*** R1.5 - Final

Issue Description Need to be able to store different types of Spaces -- Access space around doors in CS-2 (accessibility), Operable space (area where a door swing), and Operation space (space in front of an oven or stove).

Have tried to do this with AccessSpace on IfcEquipment. This proposal simply goes farther.

Proposed Solution Modify the existing Pset in IfcEquipment to include these additional space functions.

Also add AccessSpace, OperableSpace and OperationSpace to the enum for "types" of IfcSpace.

See also I-355 about a lightweight space.

Resolution

Action # 1 ***Assignee*** Liebich ***Status*** Incomplete ***Resolved in Version*** R2.0 - Beta

Investigate and propose solution.

Issue Number I - 354 ***Issue Date*** 4/30/98

Author Drogemuller ***Owner*** See ***Status*** Deferred to R2.0

Schema All Schemata ***Version*** R1.5 - Final

Issue Description Thermal boundaries (aligned to external BuildingElements) - we need to be able to define "thermal boundaries" -- boundaries to thermal zones.

Proposed Solution Assess whether SpaceBoundaries can be adapted to satisfy this requirement.

Resolution

Action # 1 ***Assignee*** Drogemuller ***Status*** Incomplete ***Resolved in Version*** R2.0 - Beta

Investigate solution proposed by JF -- does it work.

Issue Number I - 355 ***Issue Date*** 7/15/98

Author Liebich ***Owner*** See ***Status*** Deferred to R2.0

Schema	IfcProductExt	***Version***	R1.5 - Final

Issue Description We need a lightweight space object for use as AccessSpace, etc.

Proposed Solution Investigate definition of a supertype to the existing space.

Resolution

Action # 1 ***Assignee*** Liebich ***Status*** Incomplete ***Resolved in Version*** R2.0 - Beta

develop proposal for R2.0

Issue Number	***I - 356***			***Issue Date***	5/30/98
Author	See	***Owner***	See	***Status***	Resolved
Schema	IfcPropertyTypeResource	***Version***	R1.5 - Final		

Issue Description There is no real benefit to having the two subtypes of IfcPropertySet (IfcSharedPropertySet, IfcOccurrencePropertySet). In fact it causes some confusion as to when to use which.

Proposed Solution remove the two subtypes (IfcSharedPropertySet, IfcOccurrencePropertySet) and make IfcPropertySet concrete.

Resolution 15-July - agreed
During work on Psets in Aug-98: [RS] Would like to withdraw this issue as I now disagree with my initial assertion for the following reasons. There is a "1 to 1" relationship between a TypeDef and SharedPsets and a "1 to N" relationship between OccurrencePsets. This is only clearly represented by distinguishing the two with separate relationships to each.

Action # 1 ***Assignee*** See ***Status*** Eliminated ***Resolved in Version*** R1.5 - Addend

make this change for Psets in all Schemata except SharedBldgServiceElements, HVAC and FM

Eliminated

Action # 2 ***Assignee*** Forester ***Status*** Eliminated ***Resolved in Version*** R1.5 - Addend

make this change for Psets in SharedBldgServiceElements and HVAC

Eliminated

Action # 3 ***Assignee*** Yu ***Status*** Eliminated ***Resolved in Version*** R1.5 - Addend

make this change for Psets in FM

Eliminated

Issue Number	***I - 357***			***Issue Date***	9/18/97
Author	Steinmann	***Owner***	See	***Status***	Unresolved
Schema	IfcSharedBldgElements	***Version***	R1.5 - Final		

Issue Description See email discussion "URGENT ISSUE for R1.5 Addendum" which began in early June 1998. Extrusion direction for IfcWall.

The current active implementers have many problems with extrusions along the path as the norm.

Proposed Solution The current active implementers have agreed that -- if we only support extrusion for walls in a single direction in R1.5 (not 3 alternatives as proposed by STF), then that direction should be vertical.

Resolution

Issue Number	***I - 358***			***Issue Date***	7/15/98
Author	Liebich	***Owner***	Liebich	***Status***	Resolved
Schema	IfcGeometryResource	***Version***	R1.5 - Final		

Issue Description at ENTITY IfcAxis1Placement no rule enforces the location to be three-dimensional

Proposed Solution add WHERE rule WR2 that requires 3D Cartesian Point for Location.

Resolution agreed

Action # 1 ***Assignee*** Liebich ***Status*** Complete ***Resolved in Version*** R1.5 - Addend

"just do it"

Issue Number ***I - 359*** ***Issue Date*** 7/15/98

Author Liebich ***Owner*** Liebich ***Status*** Resolved

Schema IfcGeometryResource ***Version*** R1.5 - Final

Issue Description Ecco reports errors when executing FUNCTION IfcFirstProjAxis

Proposed Solution the line IF (NOT EXISTS(ZAxis) OR (NOT EXISTS(Arg)) OR (Arg.Dim <> 3) has to be replaced by IF (NOT EXISTS(ZAxis) OR ((EXISTS(Arg)) AND (Arg.Dim <> 3)), the variable Z had been deleted and its occurrence has to be replaced by Zaxis

Resolution agreed

Action # 1 ***Assignee*** Liebich ***Status*** Complete ***Resolved in Version*** R1.5 - Addend

"just do it"

Action # 2 ***Assignee*** Liebich ***Status*** Incomplete ***Resolved in Version*** R2.0 - Beta

consider writing a SEDS, since the error originates from Part42 function first_proj_axis

---

Issue Number ***I - 360*** ***Issue Date*** 7/16/98

Author Liebich ***Owner*** See ***Status*** Resolved

Schema IfcGeometryResource ***Version*** R1.5 - Final

Issue Description at ENTITY IfcAttDrivenProfileDef, the DERIVE attributes PositionToOrigin and AngleInOrigin do not add semantics, and the current computation contains errors according to the instantiation check with Ecco

Proposed Solution delete DERIVE attributes PositionToOrigin and AngleInOrigin

Resolution agreed

Action # 1 ***Assignee*** Liebich ***Status*** Complete ***Resolved in Version*** R1.5 - Addend

"just do it"

Issue Number ***I - 361*** ***Issue Date*** 7/27/98

Author See ***Owner*** See ***Status*** Resolved

Schema IfcKernel ***Version*** R1.5 - Final

Issue Description IfcProject no longer contains the "Contains" inverse relationship to IfcRelContains. All of the other containers have this inverse relationship. This must be a simple mistake?

Proposed Solution Put it back.

Resolution Agreed

Action # 1 ***Assignee*** Liebich ***Status*** Incomplete ***Resolved in Version*** R1.5 - Addend

add inverse relationship between IfcProject and IfcRelContains.

Issue Number ***I - 362*** ***Issue Date*** 7/15/98

Author Forester ***Owner*** See ***Status*** Resolved

Schema IfcUtilityResource ***Version*** R1.5 - Final

Issue Description See email thread regarding use of "globally unique object Ids".
Summary: If we are ever to enable the following:
1) exchange of patial models
2) client/server implementations that will allow checkout of model subsets
3) model servers that manage multiple models

then objects must have globally unique Ids at the object level -- not just project unique.

Proposed Solution Proposal (from J.Forester) - use Microsoft OS call for GUID
Proposals (from P.Muigg/J.Tammik) - shorten ID from 32 bytes to 20 bytes using algorithm distributed via email

Resolution 1) Agreed to use MS GUID solution for R1.5 and look for longer term solution that is not MS specific.
2) will use code for shortening GUIDs to 20 characters as provided by P.Muigg

Action # 1 ***Assignee*** Drogemuller ***Status*** Incomplete ***Resolved in Version*** R1.5 - Addend

Make necessary changes to the utility resource

Issue Number ***I - 363*** ***Issue Date*** 7/15/98

Author Poyet ***Owner*** See ***Status*** Resolved

Schema All Schemata ***Version*** R1.5 - Final

Issue Description Long Form EXPRESS is different than the Short Form EXPRESS. This creates a significant problem for developers who use the Short Form. Specifically, Explicit "ONEOF" declarations have been added in the Long Form. This is not consistent with ISO 10303-11 (definition for EXPRESS).

Proposed Solution Make them consistent and insure absolute conformance to ISO 10303-11.

Resolution The "ONEOF" declarations were added into the Short Form because of the implicit "ANDOR" in EXPRESS. As we our modeling rules only allow the use of "ONEOF", these had to be declared explicitly.

We will find a way to modify our model development toolset to insure consistency between Short Form and Long Form for IFC R2.0. For this and other reasons raised by Hartmut Steinn, we have declared that the Long Form is the only "official" form of EXPRESS for the IFC R1.5 model.

Action # 1 ***Assignee*** Liebich ***Status*** Incomplete ***Resolved in Version*** R2.0 - Beta

Work with Hartmut Stein to resolve the issues he raised.
Propose a process for developing EXPRESS for R2.0 that will result in Short Form and Long Form versions of the EXPRESS that are consistent.

Issue Number ***I - 364*** ***Issue Date*** 8/8/98

Author See ***Owner*** Liebich ***Status*** Resolved

Schema IfcPropertyTypeResource ***Version*** R1.5 - Final

Issue Description The Occurrence Pset includes a mandatory reference to a Type Def. This will not be valid in the case where an Occurrence Pset is referenced as as nested rather than directly Type Driven. In this case, the reference should be to an IfcPropertySet (not IfcPropertyTypeDef).

Proposed Solution 1) Make the relationship to IfcPropertyTypeDef optional.
2) Add an optional relationship to IfcPropertySet.

Resolution Agreed (?)

Action # 1 ***Assignee*** Liebich ***Status*** Incomplete ***Resolved in Version*** R1.5 - Addend

Make changes as proposed.

Issue Number ***I - 365*** ***Issue Date*** 8/12/98

Author IAI Implementers ***Owner*** See ***Status*** Unresolved

Schema IfcProductExt ***Version*** R1.5 - Final

Issue Description Moving the quantity related attributes that were on IfcElement (Pre-Final for R1.5) and to the Pset "Pset_ElementQuantities". Furthermore, this is inconsistent with the fact that the quantities on IfcSpace remain on the object.

Proposed Solution Move these quantities back onto IfcElement

Resolution